
LECTURE NOTES IN CONTROL AND INFORMATION SCIENCES

408

Jan H. Richter

# Reconfigurable Control of Nonlinear Dynamical Systems

## A Fault-Hiding Approach

Lecture Notes
in Control and Information Sciences 408

**Editors: M. Thoma, F. Allgöwer, M. Morari**

Jan H. Richter

# Reconfigurable Control of Nonlinear Dynamical Systems

## A Fault-Hiding Approach

Springer

**Author**

Dr. Ing. Jan H. Richter

Ruhr-Universität Bochum
Institute for Automation and Computer Control
Universitätsstrasse 150
44801 Bochum
Germany
E-mail: janhrichter@gmail.com

ISBN 978-3-642-17627-2 e-ISBN 978-3-642-17628-9

DOI 10.1007/978-3-642-17628-9

Lecture Notes in Control and Information Sciences ISSN 0170-8643



Printed in acid-free paper

5 4 3 2 1 0

springer.com

*For Diana.*

# Preface

This monograph is about control reconfiguration in fault-tolerant control, which is presently subject to vivid discussion in the literature. The monograph contributes to the control reconfiguration of nonlinear dynamical systems, which are not yet as broadly covered as linear systems. It summarises some of the outcomes of five years of research while I was a PhD associate at the Institute of Automation and Computer Control (ATP) at Ruhr-Universität in Bochum, Germany.

The progress summarised in this monograph would not have been possible without the support of several people. I would like to express my special gratitude to my supervisor Prof. Jan Lunze for his perpetual advice and support. My gratitude is extended to Maurice Heemels for introducing me to hybrid systems, to Nathan van de Wouw for sharing his insight on convergence and piecewise affine systems, and to Siep Weiland for several fruitful discussions on linear geometric control theory and the Youla parameterisation.

I also owe thanks to all my colleagues in Bochum. Special thanks go to Axel Schild for ever-interesting and humorous discussions, with whom I shared an office. Also, I thank Carsten Fritsch, Tobias Kleinert, Jörg Neidig and Philipp Planchon for frequent and very helpful discussions. My thanks for proof-reading the manuscript go to Ozan Demir, Jan Falkenhain, Daniel Lehmann, Plinio de Leon, Yannick Nke, Jörg Pfahler, Axel Schild, and Thorsten Schlage. My students Steffen Adelt, Maurice Caspar, Thomas Jakubowski, Daniel Lehmann, Christiane Leuer, Daniel Meyer, Christian Ortmann, Jesse Philipps, Thorsten Schlage, Marc Schulz, and Ye Wang have have supported applications within their Bachelor and Master theses. Mrs. Marschall has helped considerably with the production of high-fidelity figures not only for this monograph, but also with several previous publication, which I gratefully acknowledge.

Last but not least, I would like to thank Doris, Heinz, and Diana for their perpetual support.

Lüneburg, August 2010 Jan H. Richter

# Abstract

In this monograph, the fault-hiding approach to reconfigurable fault-tolerant control is extended from linear dynamical systems to two classes of nonlinear dynamical systems. Reconfigurable control is a means for improving the reliability of a controlled dynamical system that is subject to component faults, such as actuator failure or sensor failure. Reconfigurable control changes the control law after the occurrence of faults in real time and without human interaction so that the reconfigured closed-loop system continues to fulfill its function. This control adjustment step becomes necessary if the faults are so severe that they open the feedback control loop, so that, without remedial actions, the faulty system operates in open loop.

Control reconfiguration happens autonomously without human interaction, therefore it is of paramount importance that minimum invasive changes to the control law be made. This consideration favors the use of virtual sensors in the case of sensor faults, which replace the output of a broken sensor with an estimate. In the case of actuator faults, virtual actuators are used, which orchestrate the functioning actuators in order to mimic the effect of control actions of failed actuators. The advantage of the use of virtual sensors and virtual actuators as reconfiguration blocks lies in the reusability of the nominal controller in the reconfigured closed-loop system.

In this monograph, the notions of virtual sensors and virtual actuators are extended from linear dynamical systems towards Hammerstein-Wiener systems and towards piecewise affine systems. Each class of systems represents an essential aspect of nonlinear behaviour. Hammerstein systems can, in addition to expressing general nonlinear actuator characteristics, represent actuator constraints by means of input saturations. Piecewise affine systems are capable of approximating nonlinear dynamics to better accuracy and with a larger domain of validity than linear systems.

Several different reconfiguration goals of varying strength are defined and followed in this monograph, which refer to the recovery of nominal closed-loop specifications in terms of stability, setpoint tracking, and performance. The notions of virtual sensors and virtual actuators are generalised to nonlinear systems, and synthesis methods are stated and proven for the classes of Hammerstein-Wiener systems and piecewise affine systems. Depending on the system class and the

reconfiguration goal, the parameters for the virtual actuators and virtual sensors result from the solutions of equivalent output regulation problems. Sufficient conditions for the solvability of these problems are stated and compared to corresponding linear conditions. The robustness of the new methods with respect to uncertainties of the faulty plant model is shown. The methods developed in this monograph are based on several different stability and performance concepts, on linear matrix inequalities, and polytopes. Two examples are weaved into the text in order to illustrate the ideas.

The monograph concludes with descriptions of the application framework, with experimental evaluations of the new methods based on a thermofluid benchmark process implemented on a large-scale pilot plant, and a summary and discussion of open problems.

# Contents

## Part I: Control Reconfiguration Problem

## Part IV: Applications



## Part V: Appendices

# Part I
# Control Reconfiguration Problem

This part introduces and explains the reconfigurable control problem. The relevant literature is discussed, and preliminary theoretical results required to understand and to prove the main results are recalled from the literature. The reconfigurable control problem is stated in its general form, and it is suitably formalised. The fault-hiding approach is introduced, which is the general principle that all following approaches are based on. The state of the art regarding linear fault-hiding approaches to reconfigurable control is summarised.

# Chapter 1
# Introduction to Reconfigurable Control

## 1.1 Fault-Tolerant Control

Fault-tolerant control (FTC) aims at making technological systems tolerant to faults. This means that the system should fulfill its function also after the appearance of degradation or failure in its components, such as actuators or sensors. Specifically, the field is concerned with systems whose function depends on functioning *feedback control loops*. Feedback controllers process measurement information into control actions, producing the desired effect in the plant only if the involved actuators and sensors function properly. Actuators and sensors are, however, subject to inevitable faults, and fault-tolerant controllers should nevertheless maintain the system's main functionality. This monograph focusses on actuator and sensor faults.

Fault-tolerant control thus aims at giving technological systems some of the remarkable dependability properties found in biological systems. In today's complex technological systems that consist of numerous components, classical quality improvement steps at the component level become less and less sufficient for ensuring overall dependability and reliability at the aggregate system level. Single components are inevitably prone to unexpected failure, and the rising number of components in complex systems and their interactions require new strategies for ensuring reliable system operation.

System dependability defines the system to be safe and available (fulfilling its task when needed[1]). Dependability is a desirable property for several reasons of varying importance. In production and manufacturing applications, system failure potentially causes loss of financial profit due to production delays. In safety-critical control application domains, such as in the aerospace and automotive industries, system failure potentially causes personal injury or loss of life. In autonomous vehicles such as planetary exploration rovers, unmanned aircraft, or unmanned underwater vessels, reliability is mission-critical. Reliability is defined as the complementary fault probability and does not take into account maintenance strategies. System dependability is influenced by the number of involved components, by their individual

[1] See Appendix B for a glossary of the fault-tolerant control terminology.

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 3–22.
springerlink.com

reliabilities that in turn depend on the component quality properties, by maintenance schemes, and by a safety mechanism that shuts the system down into a safe state before dangerous situations are reached. Traditionally[2], a given desired level of system dependability is realised by quality assessment of the system components, by the identification of single points of failure, by the adequate installation of parallel redundant components, and by imposing and enforcing adequate maintenance schedules [161].

This traditional engineering approach to designing dependable systems seems to work well in practice, although it has the following major shortcomings:

- Quality improvement of individual components has limits imposed by budgets.
- The installation of parallel redundancies is expensive.
- Maintenance schemes tend to be conservative in the sense that the maintenance intervals are shorter than necessary in most situations.
- Maintenance and repair on demand are unavailable options in autonomous systems.

Therefore, it is vital in many applications that the system have built-in fault-tolerance properties. While quality and maintenance focus on fault avoidance and fault removal in individual components, fault-tolerant control exploits the control loops that coordinate the behaviour of these components in order to achieve fault-tolerant behaviour of the aggregate system. Fault-tolerant control complements the classical elements of dependable system design by actively responding to unexpected component failures. Through these steps, the system reliability is increased and therefore also its dependability.

The previously described methods depend on the presence of *redundancy*, which may take one of the two following forms. The term *physical redundancy* refers to the presence of multiple identical instances of components that fulfill critically important functions. Intuitive common-sense notions of redundancy usually refer to physical redundancy. However, there exists the more general notion of *analytical redundancy*, which means that multiple ways of affecting the system in a prescribed way exist, or that multiple ways for retrieving the same information about the system exist. In other words, there exist multiple mathematical relations between input variables, state variables, and output variables, so that based on a mathematical system model, an alternative relation may be used for control purposes if any relation disappears due to a fault.[3] Fault-tolerant control exploits the presence of redundancy in order to achieve the control goals in spite of faults and failures of some of its components.

Two ways of fault-tolerant control are distinguished, termed passive and active fault-tolerant control. In *passive fault-tolerant control*, the controller used during

[2] In a classical 1956 paper, John von Neumann expressed the idea that the problem of "building reliable systems from unreliable components" should be addressed in a more systematic fashion than previously [216]. This theme continues to influence electrical circuit design to date [27], and it equally applies to complex control systems.

[3] In information science, it has been long recognised that redundancy is closely linked to entropy [114].

normal operation is given a-priori robustness against system parameter changes induced by faults, such that it can directly handle a class of expected faults without actively changing the control law. Since the post-fault controller is immediately available after the occurrence of a fault, a passive fault-tolerant controller is part of most practical FTC schemes. However, the passive approach results in conservative controllers and usually leads to very limited performance. Furthermore, the passive approach fails to provide any fault-tolerance against the complete failure of main actuation or measurement elements. In order to improve the post-fault control performance and in order to cover severe faults that break the control loop, it is generally advantageous to switch to a new controller that is tailored to controlling the faulty plant, thus complementing passive FTC strategies.

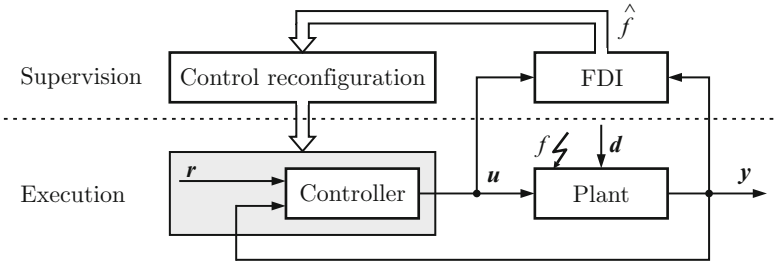


**Fig. 1.1** Active fault-tolerant control involves fault diagnosis (FDI) and control reconfiguration steps.

In *active fault-tolerant control*, the controller is actively changed after the occurrence of a fault [21]. Active FTC consists of two successive steps that each require the solution of a decision problem (Fig. 1.1):

1. Fault diagnosis (FDI), and
2. control re-adjustment.

Fault diagnosis performs three consecutive tasks [67]. It detects the presence of a fault in the system, it isolates the faulty components, and it identifies a model of the faulty system. The diagnosis component denoted as FDI in Fig. 1.1 is connected to the control input $\boldsymbol{u}$ and the measured output $\boldsymbol{y}$ of the plant. Its decision is based on the behaviour of the system as observed through the signals $(\boldsymbol{u},\boldsymbol{y})$ that are available from measurement. In summary, fault diagnosis decides which model adequately represents the faulty plant. The successive control re-adjustment step decides about a reconfigured controller that replaces the nominal controller. The re-adjustment step is called

- *fault accommodation* if the sets of manipulated and measured signals $\boldsymbol{u}$ and $\boldsymbol{y}$ remain unchanged, and if the control adjustment is limited to the controller dynamics, or
- *control reconfiguration*, if both the controller dynamics and the closed-loop structure, and possibly also the reference signal $\boldsymbol{r}$, are changed.

This monograph describes new control reconfiguration methods, therefore Fig. 1.1 shows a re-adjustment block called ”Control reconfiguration”. A control scheme that explicitly accounts for active reconfiguration of the closed-loop system during closed-loop operation is called a *reconfigurable control scheme*.

## 1.2 Reconfigurable Control

In the context of this monograph, reconfigurable control is about finding a new feedback control law, called the *reconfigured controller*, after the occurrence of faults in the system such that the reconfigured controller recovers the nominal closed-loop control goals such as stability, asymptotic tracking and performance as well as possible. The recovery usually cannot be perfect, therefore a *graceful degradation* is desirable, meaning that as many components of the vector reference signal are tracked as possible, and that the nominal performance is approximated in terms of a suitable metric. The main distinction between reconfigurable control and fault accommodation is that reconfigurable control includes modifications of the closed-loop signal structure, which are excluded in fault accommodation. In other words, the use of different input and output signals is allowed in reconfigurable control.

Reconfiguration in fault-tolerant control exploits the presence of redundancy in the controlled system. Reconfigurable control is based on models of the fault-free and the faulty system, where the latter is provided by the fault diagnosis component [21] or by self-diagnosing actuators and sensors [49]. The model of the faulty plant must express all redundancies so that automatic reconfiguration methods are enabled to exploit them. An example for controller reconfiguration is shown in Fig. 1.2 for a failure of the second actuator.

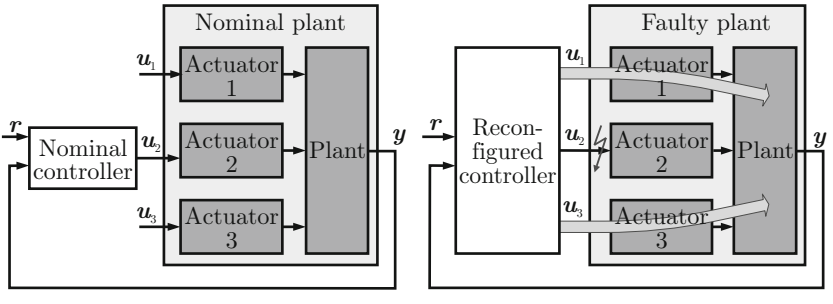


**Fig. 1.2** Control reconfiguration after actuator failure re-routes the control action from faulty to healthy actuators.

The re-routing of inputs around broken actuators is the key problem of reconfigurable control after actuator failure, as shown in Fig. 1.2, where Actuator 2 fails. The reconfigured controller uses Actuator 1 and Actuator 3 to mimic the effect of

Actuator 2 through different plant dynamics. In order to achieve successful reconfiguration, the Actuator 1 and Actuator 3 must together provide analytically redundant alternatives for the functionality provided by the failed Actuator 2. Likewise, the reconfiguration problem after sensor faults consists in finding a control law and a suitable measurement vector to recover control over the variables of interest. The problem consists in finding suitable alternative inputs and outputs avoiding the broken ones, in other words, in adjusting the sets of control inputs and measured outputs used by the reconfigured controller, as well as the dynamics of the reconfigured controller.

Reconfigurable control must at least ensure that the reconfigured closed-loop system is stable in a suitable sense. Furthermore, it is desirable to recover the nominal closed-loop tracking and performance properties as far as possible. The exact recovery of these properties is typically possible only in the presence of physical redundancy in the system. In numerous technological systems, physical redundancy is not available due to its cost. The appearance of actuator faults typically turns the nominal system into an underactuated system, whereas the nominal system may be either fully actuated or underactuated.

From a broader perspective, reconfigurable control is a general control adjustment technique. Generally speaking, the need for reconfigurable control arises whenever the controlled plant abruptly undergoes substantial structural changes. Such changes in the plant can occur when certain components are switched off for scheduled maintenance, or they occur as a consequence of abruptly appearing fault effects. This monograph discusses reconfigurable control within the context of fault-tolerance. Reconfigurable control after actuator failures is related to *control allocation* as follows. The control allocation problem arises if a desired forcing action can be realised by means of several combinations of the available actuators [106]. Control allocation problems typically arise in overactuated systems such as aircraft, where multiple control surfaces are capable of producing the same aerodynamical forces and moments. Since faults typically result in underactuated systems, the reconfigurable control problem is more difficult than the control allocation problem. The control allocation problem is contained in the reconfigurable control problem as a special case. Due to the general formulation of the reconfiguration problem, the results obtained in this monograph are easily transferred to other application domains, such as control allocation.

## 1.3 Key Aspects Specific to Reconfigurable Control

The following aspects characterise the reconfigurable control problem and illustrate why the synthesis of reconfigurable controllers is much harder than nominal controller syhesis.

- The reconfigured controller must be found *online* and *autonomously*. Namely, the typical design cycle involving test simulations and engineering judgement of the responses' adequateness is not available in autonomous reconfigurable control. To satisfy autonomy, any design freedom available in reconfigurable

control synthesis methods must be either automatically assigned, or systematically and a priori eliminated due to the lack of interaction with control engineers.
- The computations must be completed in *real-time*, namely fast relative to the time constants that govern the system behaviour. Otherwise, the reconfiguration delay becomes unacceptably large. For this reason, only those controller synthesis methods are usable with reconfigurable control that permit efficient implementation in an autonomous computer program.
- The model of the faulty plant provided by the fault diagnosis component is typically uncertain. Thus, the reconfiguration should make minimum-invasive changes to the control law, and it should be *robust* against uncertainties in the model of the faulty plant provided by the diagnosis component. Especially if the faults affect few components of a large-scale plant, then the control actions to intact components given by the nominal controller may still be valid.

The reconfigurable control methods described in this monograph are inspired by these considerations and have been developed to meet these requirements. It is also worthwhile noting at this point that in order to achieve successful model-based control reconfiguration, the underlying models must represent all redundancies that are present in the physical system. Otherwise, these redundancies cannot be exploited in a model-based way.

## 1.4 Contributions and Structure of This Monograph

The reconfigurable control approach developed in this monograph is based on the idea of placing a reconfiguration block $\Sigma_R$ in between the nominal controller $\Sigma_C$ and the faulty plant $\Sigma_{Pf}$ at reconfiguration time (see Fig. 1.3). The reconfiguration block hides the fault from the controller. In other words, the reconfigured plant $\Sigma_{Pr}$ seen from the signal pair $(\boldsymbol{u}_c, \boldsymbol{y}_c)$ must have the same input/output behaviour as the nominal plant $\Sigma_P$ seen from the signal pair $(\boldsymbol{u}_c, \boldsymbol{y})$. The reconfiguration block contains a virtual sensor (an observer-like system) and a virtual actuator (a dual observer-like system) in the general case. Apart from fault-hiding, the reconfiguration block must achieve as many of the following goals as possible, which are formalised in later chapters. The reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_R, \Sigma_C)$

1. must be stable,
2. should recover the tracking properties of the nominal closed-loop system $(\Sigma_P, \Sigma_C)$,
3. should recover the performance properties of the nominal closed-loop system $(\Sigma_P, \Sigma_C)$.

This idea is called the *fault-hiding principle*. It opens the way for minimum-invasive changes of the controller. Its goal consists in the *recovery* of the mentioned nominal closed-loop properties, therefore, the nominal closed-loop properties (such as overshoot and settling time) need not be precisely specified for the purpose of

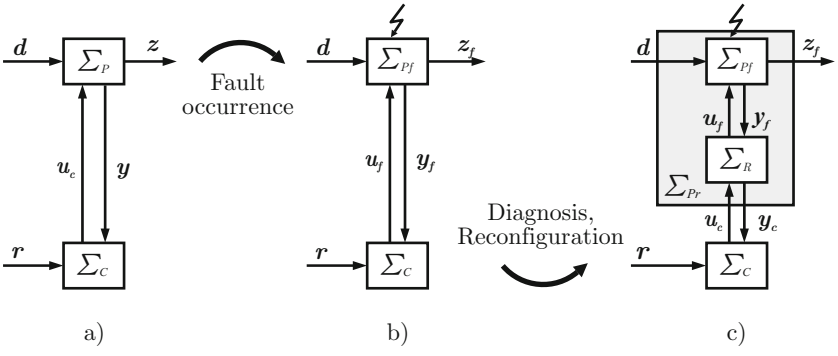


**Fig. 1.3** Fault-hiding approach to reconfigurable control: a) nominal closed-loop system, b) faulty closed-loop system prior to diagnosis and reconfiguration, c) reconfigured closed-loop system.

reconfigurable control. Furthermore, the reconfiguration problem is completely separated from the controller in the fault-hiding approach, and its solvability analysis only refers to system properties, instead of properties of the nominal controller. All the reconfiguration solutions obtained in this monograph interoperate with arbitrary nominal controllers.

Figure 1.4 illustrates that after the occurrence of a fault at time $t_f$, the system deviates from its specification (dashed). At time $t_D$, the fault is diagnosed, and the determination of the reconfigured controller proceeds, finishing at time $t = 0$, when the reconfigured controller is available. At that time, the faulty plant being in the state $\boldsymbol{x}(0) = \boldsymbol{x}_0$. In the ideal case of perfect reconfiguration (or repair of the system), the reconfigured system trajectories would follow nominal dynamics (solid in Fig. 1.4). Realistically, the reconfiguration is not ideal at the state level, so that the reconfigured system state trajectories differ from the nominal ones (dotted in Fig. 1.4). The difference state is called $\boldsymbol{x}_\Delta$, see Fig. 1.4 and will be central to the ideas presented in this monograph.

The fault-hiding approach is extended from linear systems to two special classes of nonlinear dynamical systems: Hammerstein-Wiener systems and piecewise affine

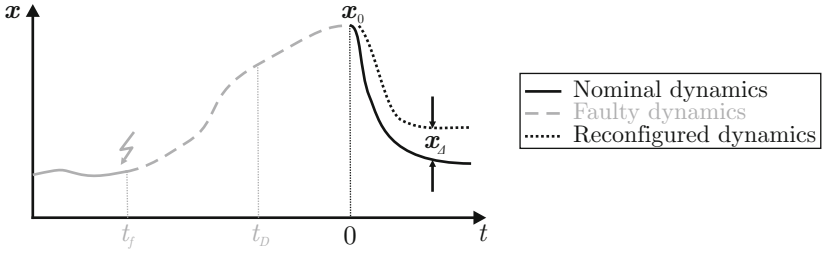


**Fig. 1.4** The reconfiguration problem starts at the initial time $t = 0$ and the initial condition $\boldsymbol{x}_0$.

systems. *Hammerstein-Wiener* systems consist of linear dynamics and static functions distorting the control inputs and measured outputs. In particular, these systems can represent actuator saturation, which occur in most practical applications and typically pose a challenge to stabilising control. In reconfigurable control after actuator failure, typically the actuation power of lost actuators must be suitably distributed to the remaining actuators. Consequently, fewer actuators have to generate the same control effect, which makes the activation of actuator saturations likely and the recovery of stability a particular challenge. For this reason, the development of control reconfiguration theory for Hammerstein-Wiener systems is of considerable practical relevance. *Piecewise affine* systems consist of a collection of affine dynamics (linear plus offset), each of which is active in a particular region of the state space. Piecewise affine systems can well approximate certain important classes of nonlinear dynamics. Nonlinear dynamics become important as the system ranges through a large part of the entire operating region. Such operation arises in startup procedures, but also when the system departs from its operating point before the fault can be detected and the reconfigured controller is activated. Thus, the extension of control reconfiguration theory towards nonlinear dynamics is important. In this monograph, continuous piecewise affine systems are considered.

For both classes of nonlinear systems, stability and tracking recovery problems are solved. For saturated systems, the performance recovery problem is also solved. The solutions are extensions of the virtual actuator and the virtual sensor, which are known from linear systems, towards Hammerstein-Wiener and piecewise affine systems. As an example, the linear virtual actuator $\Sigma_A$ is shown in Fig. 1.5.

The virtual actuator consists of a dynamical system that keeps track of the state deviation $\boldsymbol{x}_\Delta$ from nominal that is due to the actuator faults. This deviation is used to determine fault compensation action $\boldsymbol{M}\boldsymbol{x}_\Delta$ and a measurement correction $\boldsymbol{y}_\Delta$. The linear virtual actuator is based on the superposition principle and the linear separation principle. The extension of the virtual actuator and the virtual sensor in this monograph overcomes the lack of both principles in nonlinear systems, where suitable alternatives are used. All algorithms are suitable for autonomous implementation, namely all algorithms work without user interaction. This monograph is structured as follows.

Part I formulates and explains the reconfigurable control problem. The mathematical background common to most chapters is briefly reviewed in Chapter 2. The definitions of general reconfigurable control problems for nonlinear systems based on the nominal and faulty system models are stated in Chapter 3. Fault-hiding solutions to these problems for linear systems are recalled and extended in Chapter 4.

Part II covers the extension of the fault-hiding principle towards Hammerstein-Wiener systems. The classes of nominal and faulty Hammerstein-Wiener systems are introduced in Chapter 5, where also specific reconfiguration problems are formulated. The stability recovery problem is solved for Hammerstein-Wiener systems and combined actuator and sensor faults in Chapter 6. The problem of recovering the nominal closed-loop setpoint tracking properties is solved for saturated systems subject to actuator faults in Chapter 7. The additional recovery of the nominal

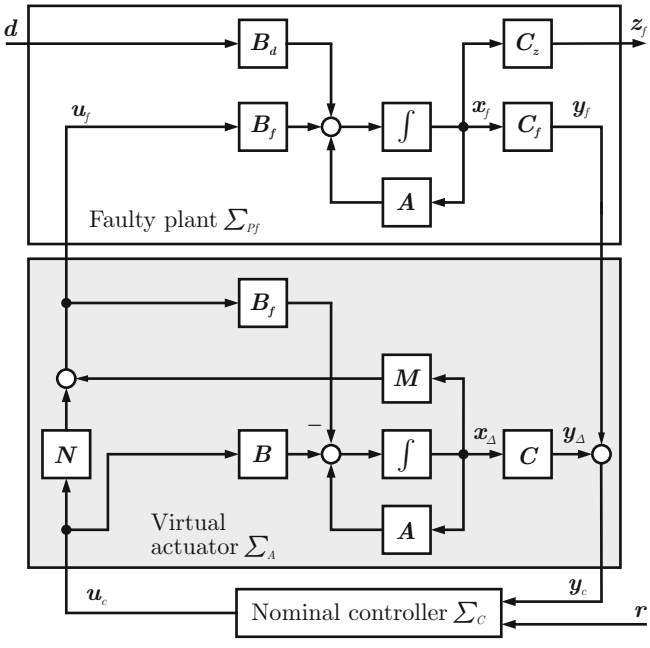


**Fig. 1.5** The linear virtual actuator shown above is extended towards Hammerstein-Wiener systems and to piecewise affine systems in this thesis.

closed-loop performance properties for saturated systems subject to actuator faults is solved in Chapter 8.

Part III describes the extension of the fault-hiding principle towards piecewise affine systems. The classes of nominal and faulty piecewise affine systems are introduced in Chapter 9, where specific reconfiguration problems for piecewise affine systems are formulated as well. The stability recovery problem is solved in Chapter 10. The additional recovery of the nominal closed-loop tracking properties is provided in Chapter 11. Robustness issues arising from the interconnection of piecewise affine reconfiguration blocks with nonlinear systems are discussed in both chapters.

Part IV shows applications of the developed methods. The general application framework is summarised in Chapter 12. The experimental fault-tolerant control of a thermofluid process is shown in Chapter 13. The monograph concludes with a discussion of the results and open problems in Chapter 14.

Part V contains appendices that define acronyms and mathematical symbols (Appendix A), that explain fault-tolerant control terminology in a glossary (Appendix B), that define basic notions of linear geometric control (Appendix C), that provide the technical proofs that do not fit well into the main text (Appendix D), and that provide models of the thermofluid process used as an application example in this monograph (Appendix E).

Throughout this monograph, definitions of key concepts and theorems that represent major results are placed inside black frames to distinguish them from less central definitions and results. The relevant literature is discussed in bibliographic notes at the end of chapters, where applicable.

## 1.5 Running Examples

### *Ship Control*

A powered ocean vessel is used as the first running example. The considered dynamics concern the (forward) surge velocity $v$, the (sideways) sway velocity $w$, the yaw angular rate $r$, and the heading $\psi$. All velocities are defined relative to a frame of reference that is attached to the ship and relative to the water, which means that the surge velocity is defined along the center axis (Fig. 1.6).

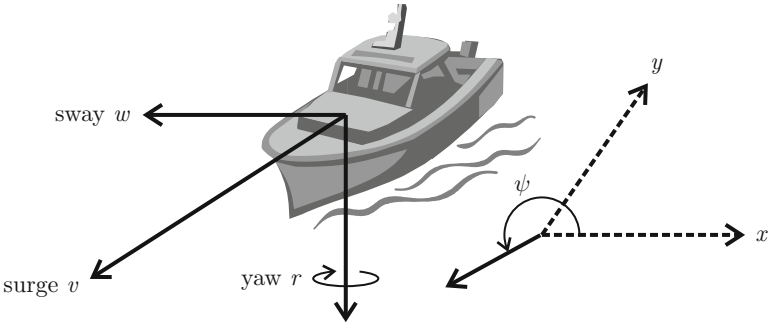


**Fig. 1.6** Ship with local and earth-fixed reference frames.

The ship has two fixed-angle thrusters that exert the forces $u_1$ and $u_2$ in the range $\underline{u}_1 \le u_1 \le \overline{u}_1$, $\underline{u}_2 \le u_2 \le \overline{u}_2$ and that are mounted symmetrically left and right of the ship centerline at the distance $b$. The rudder is used to apply a yaw moment $u_3$ in the range $\underline{u}_3 \le u_3 \le \overline{u}_3$. The thruster forces $u_1$ and $u_2$ and the rudder moment are the available control inputs to the system. All units are metric, thus forces are given in Newton (N), torques are given in Newtonmeter (Nm), distances are given in Meters (m), angles are given in Radian (rad), and time is given in Seconds (s). Due to this convention, the units are frequently omitted.

Apart from the intentional propulsion forces and steering moments, the ship's motion is affected by wind forces given in ship coordinates by $a_v$ along the surge direction and $a_w$ along the sway direction. The wind/force model neglects the variations of a ship's cross-section when seen from different directions.

With the mentioned simplifications and assumptions, the ship motion is described by the set of equations

$$\dot{v}(t) = \frac{m_{22}}{m_{11}} w(t) r(t) - \frac{d_{11}}{m_{11}} v(t) + \frac{1}{m_{11}} (u_1(t) + u_2(t)) + a_v(t) \tag{1.1}$$

$$\dot{w}(t) = -\frac{m_{11}}{m_{22}} v(t) r(t) - \frac{d_{22}}{m_{22}} w(t) + a_w(t) \tag{1.2}$$

$$\dot{r}(t) = \frac{m_{11} - m_{22}}{m_{33}} v(t) w(t) - \frac{d_{33}}{m_{33}} r(t) + \frac{1}{m_{33}} \left( \frac{b}{2} (u_1(t) - u_2(t)) + u_3(t) \right) \tag{1.3}$$

$$\dot{\psi}(t) = r(t), \tag{1.4}$$

where the parameters $m_{ii} > 0$, $i = 1,2,3$ are given by the ship inertia and added mass effects [160]. The parameters $d_{ii} > 0$ are given by hydrodynamical damping. In order to obtain a solution of the system (1.1)–(1.4) starting from the time $t_0$, the initial surge velocity $v(t_0) = v_0$, the initial sway velocity $w(t_0) = w_0$, the initial yaw rate $r(t_0) = r_0$, and the initial heading $\psi(t_0) = \psi_0$ must be provided.

The available measurements are the surge velocity $v$ (from a speedometer), the yaw rate $r$ (from a gyrometer), and the heading $\psi$ (from a compass), thus the equations for the measured outputs are

$$y_1(t) = v(t) \tag{1.5}$$

$$y_2(t) = r(t) \tag{1.6}$$

$$y_3(t) = \psi(t). \tag{1.7}$$

The relevant controlled variables are the surge velocity $v$ and the heading $\psi$, thus the equations for the controlled outputs are

$$z_1(t) = v(t) \tag{1.8}$$

$$z_2(t) = \psi(t). \tag{1.9}$$

The performance requirements are stability of the ship motion in the sense that bounded inputs cause bounded state variables, asymptotic reference tracking for surge velocity $v$ and heading $\psi$, and 10% overshoot limits on these two variables. Although the ultimate goal consists in position trajectory following, a cascaded autopilot scheme with a fast inner velocity/heading control loop and a slower outer position control loop with reference trajectory generator is assumed to be present. For the purpose of studying fault-tolerant control, only the inner loop for velocity and heading control is considered.

The investigations regarding system analysis, nominal control, and fault-tolerant control will, therefore, be mostly done with respect to the surge, sway, and yaw velocities. Sometimes, however, the ship's motion will be illustrated with respect to an earth-fixed reference frame with the coordinates $x$ and $y$. The relationship between the ship-fixed reference frame and the earth-fixed reference frames is given by the nonlinear kinematics

$$\dot{x}(t) = v(t)\cos(\psi(t)) - w(t)\sin(\psi(t)) \tag{1.10}$$

$$\dot{y}(t) = v(t)\sin(\psi(t)) + w(t)\cos(\psi(t)), \tag{1.11}$$

and the relationship between the wind forces $a_v$ and $a_w$ in the ship's coordinates and the wind forces $a_x$ and $a_y$ in earth-fixed coordinates is given by the geometric relationship

$$a_v(t) = a_x(t)\cos(\psi(t)) + a_y(t)\sin(\psi(t)) \tag{1.12}$$

$$a_w(t) = -a_x(t)\sin(\psi(t)) + a_y(t)\cos(\psi(t)). \tag{1.13}$$

The ship parameters used in this monograph are defined in Table 1.1. The model reflects a small-scale supply vessel [160].

**Table 1.1** Parameters of the small-scale ship.

| Parameter | Value | Saturation limit | Value |
|---|---|---|---|
| $m_{11}$ | 19 | $\underline{u}_1$ | -1 |
| $m_{22}$ | 35.2 | $\overline{u}_1$ | 1 |
| $m_{33}$ | 4.2 | $\underline{u}_2$ | -1 |
| $d_{11}$ | 4 | $\overline{u}_2$ | 1 |
| $d_{22}$ | 10 | $\underline{u}_3$ | -1 |
| $d_{33}$ | 1 | $\overline{u}_3$ | -1 |
| $b$ | 0.1 | | |

The studied faults are the following:

- $f_1$: failure of the yaw rate gyro sensor,
- $f_2$: blockage of the rudder producing a constant yaw moment,
- $f_3$: floating rudder producing zero yaw moment,
- $f_4$: reduction of left thruster force range to 60%, in other words, $u_1 \in [-0.6; 0.6]$.

The particular problem to be solved in this monograph consists in finding reconfigured controllers automatically and online. The practical goal that is used as a basis for evaluating the reconfiguration success consists in circumnavigating an obstacle (Fig. 1.7). Completing this task requires two changes of the heading in opposite directions in a sufficiently precise manner, and a positive velocity must be maintained in order to move past the obstacle.

It is intuitively clear that the control goals are achievable after all fault scenarios. The loss of yaw rate information can be replaced with an estimate based on the heading measurement. The rudders can be replaced because two thrusters are installed that can be used to produce equivalent yaw moments. A thrust reduction in one of the thrusters can be accommodated by increasing the thrust command within its physical bounds, and a yaw moment produced by asymmetrical thrust can, to some extent, be compensated by setting a small rudder angle. In summary, given enough time, a tailored controller can be designed for every fault scenario.

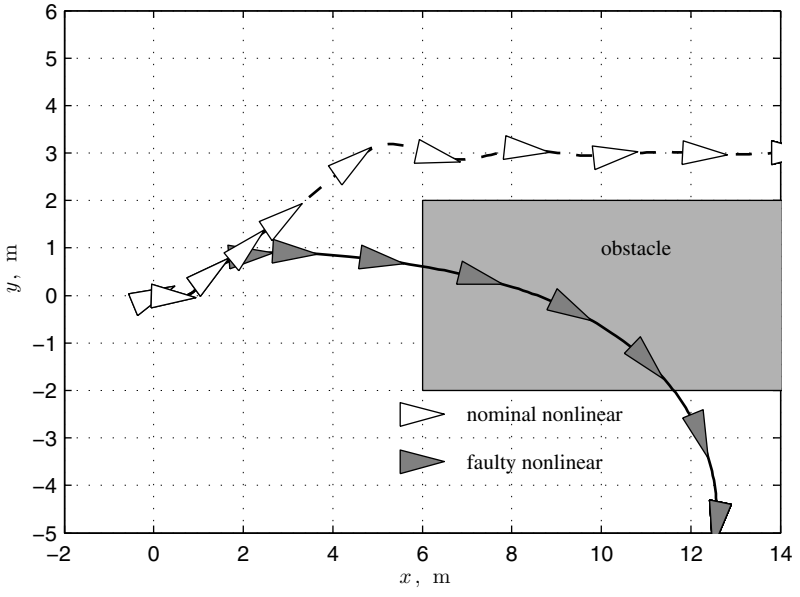


**Fig. 1.7** The nominal ship avoids the obstacle, while the ship subject to fault $f_3$ runs into the obstacle.

## *Two-Tank System Control*

The two-tank system defined as a benchmark problem in earlier work is used as a second running example. The two-tank system is pictorially shown in Fig. 1.8. It is particularly suitable for the use of piecewise affine models, since its physical setup induces natural switching, as it will be obvious from the model below.

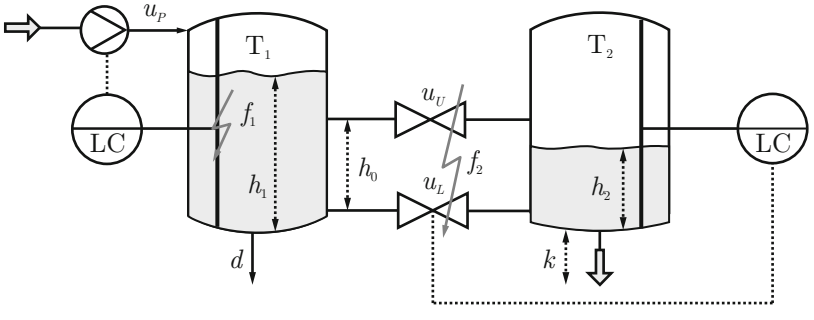


**Fig. 1.8** Two-tank system with nominal control loops.

The plant consists of tanks $T_1$ and $T_2$ with levels $h_1$ and $h_2$, respectively. The tanks are interconnected by valves accessed through the inputs $u_L$ and $u_U$, where $T_1$

is filled by means of the pump with the input $u_P$ (Fig. 1.8). A model of the system of coupled tanks is obtained from mass balances and the laws of turbulent flow, where the signal $s(t) = \text{sign}(h_1(t) - h_2(t))$ describes the direction of the flow from $T_1$ to $T_2$:

$$\begin{cases} \dot{h}_1(t) &= \frac{1}{A\rho}\Big(-\rho c_V s(t)\sqrt{2g|h_1(t)-h_2(t)|}u_L(t) \\ &\quad -\rho c_V s(t)\sqrt{2g|\max(h_1(t),h_0)-\max(h_2(t),h_0)|}u_U(t) \\ &\quad +c_{P1}+c_{P2}u_P(t)\Big) \\ \dot{h}_2(t) &= \frac{1}{A\rho}\Big(\rho c_V s(t)\sqrt{2g|h_1(t)-h_2(t)|}u_L(t) \\ &\quad +\rho c_V s(t)\sqrt{2g|\max(h_1(t),h_0)-\max(h_2(t),h_0)|}u_U(t) \\ &\quad -\rho c_O\sqrt{2g(h_2(t)+k)}\Big) \\ \boldsymbol{y}(t) &= \begin{pmatrix} h_1(t) \\ h_2(t) \end{pmatrix} \\ z(t) &= h_2(t). \end{cases} \tag{1.14}$$

In the model, $A$ denotes the cross-section area of both tanks, $k$ is the head of the outflow pipe, $h_0$ is the elevation of the upper connection pipe and the upper valve, $g$ is the acceleration constant, $\rho$ is the density of water, $c_{P1} < 0$ is a pump bias that represents a dead zone of the pump, $c_{P2}$ is a pump coefficient, and $c_V$ is a valve flow coefficient. Both inputs are normalised in the interval $[0;1]$. Numerical values for the model parameters are defined in Table 1.2. The measured output $y$ is used for control purposes, whereas the relevant regulated output $z$ reflects the main interest in the right tank fluid level. The system undergoes *switching* of its dynamics whenever the fluid levels in the tanks cross the elevation $h_0$ of the upper connection valve.

**Table 1.2** Parameters of the two-tank system.

| Parameter | Value | Saturation limit | Value |
|---|---|---|---|
| $g$ | 9.81 m/s$^2$ | $\overline{u}_L$ | 1 |
| $\rho$ | 998 kg/m$^3$ | $\overline{u}_U$ | 1 |
| $A$ | 0.0154 m$^2$ | $\overline{u}_P$ | 1 |
| $c_{P1}$ | −0.0140 m/s | $\underline{u}_L$ | 0 |
| $c_{P2}$ | 0.1251 m/s | $\underline{u}_U$ | 0 |
| $c_V$ | $2 \cdot 10^{-5}$ m$^3$ | $\underline{u}_P$ | 0 |
| $h_0$ | 0.3 m | | |
| $k$ | 0.05 m | | |

The two-tank system is controlled by two linear decentralised controllers

$$\begin{pmatrix} u_P(t) \\ u_L(t) \\ u_U(t) \end{pmatrix} = \begin{pmatrix} 50(r_1(t)-y_1(t)) + 4\int_0^t (r_1(\tau)-y_1(\tau))\mathrm{d}\tau \\ 50(r_2(t)-y_2(t)) + 4\int_0^t (r_2(\tau)-y_2(\tau))\mathrm{d}\tau \\ 0.8 \end{pmatrix}.$$

The controlled quantities are the fluid levels $h_1$ and $h_2$, for which the control aims are stability and regulation to a given setpoint. The considered faults are abrupt and non-transient:

- $f_{a1}$: failure of the lower valve $u_L$ ($u_{f,L}(t) = 0$ for $t > t_{fa1}$) at fault time $t_{fa1} = 20$ s,
- $f_{a2}$: gain reduction for the upper valve ($u_{f,U}(t) = 0.2u_U(t)$ for $t > t_{fa2}$) at fault time $t_{fa2} = 35$ s,
- $f_s$: outage of the level sensor for $h_1$ ($y_{f,1}(t) = 0$ for $t > t_{fs}$) at time $t_{fs} = 40$ s.

The plant is perturbed by reference steps $r_1(t) = 0.15$ m for $t \leq 30$ s and $r_1(t) = 0.45$ m for $t > 30$ s for the level $h_1$ as well as $r_2(t) = 0.05$ m for $t \leq 100$ s and $r_2(t) = 0.08$ m for $t > 100$ s for the level $h_2$. The steps drive the process through a large operating range, and thus realistically describe a startup procedure. A non-modelled outflow of tank $T_1$ represents a disturbance $d$. Note that the fault breaks the loop at several points and the reconfiguration method must change the control loop structure to meet the control objectives.

## 1.6 Survey of Fault-Tolerant Control

Fault-tolerant control with a focus on the control aspects is treated in depth in the survey [87], in the tutorial [120], as well as the research monograph [21]. Comprehensive commented bibliographies are available in [249, 250].

### *Passive FTC*

A *passive fault-tolerant control scheme*, where the control law is never changed, was described for linear systems in [206] based on simultaneous stabilisation techniques. An optimal sensor selection scheme to achieve passive fault-tolerance with respect to sensor faults is described in [103]. An observer-based output-feedback controller that is robust against sensor faults is available in [52]. Robust output-feedback model predictive control with soft state and hard input constraints for linear systems has been recently studied in [108]. A robust controller for linear parameter-varying systems that also estimates additive faults is described in [221]. Passive fault-tolerant control of nonlinear systems that does not explicitly distinguish diagnosis and reconfiguration was presented in [26] and extended in [18], which is based on the use of a control Lyapunov function. An idea that complements fault-tolerant control techniques is the development of high-redundancy actuators that are inspired by biological muscles and designed for graceful performance degradation [44, 203].

### *Fault Diagnosis*

Fault diagnosis methods seek to find out whether or not a system is faulty. Fault diagnosis is part of every *active fault-tolerant control scheme* and the success of subsequent control re-adjustment steps depends on the reliability of its diagnostic

results. Introductions to fault diagnosis methods are available in the books [21, 48, 67, 83].

Fault diagnosis can be signal-based or model-based, and model-based approaches are often founded on consistency tests for the I/O data on system models [21] or on consistency tests for identified system parameters [83]. Conceptually, consistency-based fault diagnosis is closely related to *behaviours* as defined by Willems [224]. Roughly speaking, the behaviour of a system with inputs $\boldsymbol{u} \in \mathcal{U} \subset \mathcal{L}_1^{loc}(\mathbb{R}^m)$ and outputs $\boldsymbol{y} \in \mathcal{Y} \subset \mathcal{L}_1^{loc}(\mathbb{R}^q)$ is defined to be the set of all pairs $(\boldsymbol{u}(t), \boldsymbol{y}(t))$ that are compatible with the system dynamics, where $\mathcal{U}$ and $\mathcal{Y}$ denote the input and output signal spaces: $\mathcal{B} \subset \mathcal{U} \times \mathcal{Y}$. In the behavioural setting, it is straightforward to explain the basic principles of fault diagnosis and also the conditions determining whether or not fault diagnosis attempts can be successful. The nominal (fault-free) system, denoted by the fault case $f_0$, is characterised by a specific behaviour denoted by $\mathcal{B}_0$. With every possible fault scenario $f_i \in \mathcal{F} = \{f_0, f_1, \ldots, f_F\}$ from a finite set $\mathcal{F}$ of possible fault scenarios, a specific behaviour $\mathcal{B}_i$ is associated. The restriction to a finite fault set is often called the *closed-world assumption*, meaning that no other faults are possible. Statements about the completeness of fault diagnosis algorithms frequently rely on this assumption. Typical example behaviours of different fault scenarios are illustrated in Fig. 1.9.

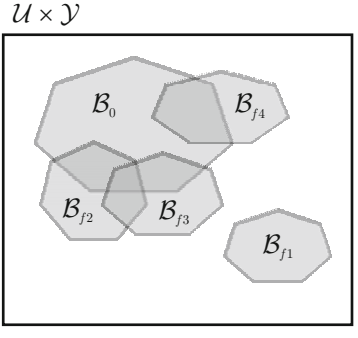


**Fig. 1.9** Behaviours of nominal and faulty systems.

The figure supports the following important conclusions about *diagnosability* of dynamical systems. If two behaviours belonging to different fault scenarios are identical, then these fault scenarios are not distinguishable. If their intersection is empty, then the two fault scenarios are perfectly distinguishable (the fault case triplets $f_1$, $f_3$, $f_4$ and $f_1$, $f_2$, $f_4$ are perfectly distinguishable in Fig. 1.9). If their behaviours overlap, as they do in most cases, then there exist signals that permit distinction between the fault scenarios (faults $f_2$ and $f_3$ in Fig. 1.9). The latter case is commonly considered to characterise a diagnosable system. From these considerations, it is clear that the diagnosis success also depends crucially on the chosen input sinals. This dependence is intimately linked to the persistence of excitation condition encountered in system identfication. It implies that test signal synthesis is a nontrivial and important

issue in fault diagnosis (*experiment design* is the parallel problem in system identification). Every complete[4] consistency-based fault diagnosis method starts with the initial fault candidate set $\mathcal{F}_c(0) = \mathcal{F}$ and iteratively excludes fault scenarios that are inconsistent with the observations. A fault $f_k$ is uniquely diagnosed at the step $j$ if $\mathcal{F}_c(j) = \{f_k\}$. In general, it is not possible to uniquely distinguish between all faults due to the overlap in their behaviours, even if full freedom in the choice of the inputs is available. Based on the retrievable information, it is possible to define a smallest fault candidate set $\mathcal{F}^\star$ that contains the true fault. Practical algorithms are evaluated by comparing their resultant fault candidate sets to $\mathcal{F}^\star$.

The consistency-based approach is available for networked discrete-event systems modelled by input/output automata [140, 189], for timed input/output automata [208], for discretely controlled continuous systems [116], for linear systems subject to biased uncertain measurements [163], and for nonlinear systems with the same uncertainties [225]. All mentioned algorithms possess the completeness property. Often, a residual is generated by means of observers or Kalman filters and compared to a threshold [209, 252]. The placement of sensors under observability and redundancy constraints is studied in [126].

This monograph addresses the control reconfiguration problem for nonlinear systems, therefore fault diagnosis methods for nonlinear systems are required. An actuator fault detection method for uncertain input-affine single-input single-output systems based on neural approximation models is described in [194]. Further fault diagnosis methods for nonlinear systems are available in [9, 248].

### *Control Reconfiguration*

There is a general consensus that actuator faults are more difficult to treat than sensor faults. After sensor faults are detected, it remains to mask their effect in the closed-loop system, typically by replacing the measurements with estimates obtained based on state-observation principles [229]. The recovery methods from sensor faults described in this monograph are in accordance with this general idea.

In reconfigurable control, the *linear model-matching approach* dominates the field, where the nominal controller is replaced by a new controller that is tailored to the faulty plant and either synthesised online, or picked from a bank of pre-designed fault-case controllers. The original approach was based on the comparison of the nominal and reconfigured closed-loop system matrices [33] but lacked a stability guarantee. This drawback was removed in [64]. Recently, the model matching idea is usually based on linear eigenstructure assignment [7], on extensions of the classical linear pseudoinverse method [98, 201], on robust $H_\infty$-control mixers [242], and on adaptive control principles [37]. The perfect model following technique has been considered as well in [65]. The disadvantage of these methods consists in the prescription of special structures for the reconfigured controller, discarding the nominal controller from the loop.

[4] A fault diagnosis algorithm is said to be *complete* under the closed-world assumption if the true fault is never excluded from the set of fault candidates.

A fault accommodation approach for nonlinear systems is described in [86]. A control reconfiguration approach for non-minimum phase nonlinear systems with control re-allocation and reference adjustment is described in [17]. Recently, a nonlinear fault accommodation technique based on nonlinear parametric estimation has been suggested, where the faults are viewed as multiplicative external inputs [57]. For switched and hybrid systems, adaptive schemes [236], observer-based switching schemes based on multiple Lyapunov functions [237], and output feedback controller redesign [181] have been developed, however based on additive fault models. In [233], fault-tolerance analysis based on global passivity is addressed, however without synthesis procedures. Periodic systems were considered in [235]. A hybrid controller approach based on hybrid automata models and verification techniques has been described in [153, 241]. A sensor fault accommodation technique based on bond graphs is available in [238].

*Model-predictive control* has been used as a basis for the reconfigurable control of linear, piecewise affine, and fuzzy systems in [127, 130, 184, 213], and an optimisation technique based on hybrid automata has been proposed in [210]. Further ideas for the reconfigurable control of hybrid systems are based on automaton abstractions [121, 122]. However, these methods require considerable online computational power. The suitability of internal model control for the accommodation of actuator faults, sensor faults, and internal faults has been studied in [77], where fault accommodation is achieved by changing the internal plant model and its inverse to let them reflect the faults. A so-called generalised internal model control scheme was developed in [34] and extended to handle model uncertainties in [240].

Techniques that explicitly *combine fault diagnosis and control reconfiguration* are difficult to achieve. Although the problem is still considered unsolved, it is tackled from various sides described in the following literature. The uncertainties of fault diagnosis are explicitly taken into account in [243]. A probabilistic approach that takes into account missed detections and false alarms has been presented in [128], which comes at the cost of very high computational complexity that limits the applicability to offline synthesis and to usage within controller banks. A general architecture for integrated residual filter design for fault diagnosis and control adjustment is suggested in [40], including a bump-less switching scheme between the nominal controller and the reconfigured controller. Joint fault diagnosis and reconfiguration approaches for actuator and sensor faults based on invariant set theory and controller banks are described in [129, 146, 147] and applied to induction motor control in [196]. In these approaches, it is required that any combination of faulty plants and fault-case controllers yields a stable reconfigured closed-loop system, which is a strong assumption. Subspace predictive control is a recently introduced combined parameter identification and controller synthesis approach that bears potential for solving the simultaneous fault diagnosis and control adjustment problem [50]. The application of unfalsified control to fault-tolerant control is a relatively new area that is still in its infancy [82]. It provides a systematic means for selecting one controller from a set of finitely many candidate controllers. The fault diagnosis and reconfiguration problem has been studied for bimodal piecewise affine systems in [138].

*Adaptive control* is typically applied to respond to incipient faults that represent gradual component degradation [2, 25]. In so far, adaptive control is readily well-suited for fault accommodation purposes. Adaptive control can be extended for reconfigurable control purposes. A fault-tolerant self-tuning adaptive control method with integrated fault detection is described in [212]. An adaptive fault compensation scheme for actuator faults is described in [251].

*Intelligent control* is concerned with learning from experience in order to address unanticipated faults. Learning control combines ideas from classical control and computer science, especially with ideas of artificial intelligence, to adapt to changing environments and to profit from past experience. Learning techniques such as neural networks, expert systems, and general problem solvers are used to store knowledge about past fault situations and the success of certain responses [51, 204]. An adaptive fault-compensation scheme based on neural network model and internal model control for nonlinear systems is described in [61], where neural networks are used to learn the nominal and faulty plant models. The concepts of adaptive control, uncertainties, and fuzzy plant models have been combined to achieve actuator fault tolerance in [247].

*Combined studies of reconfigurable and networked control* are recently emerging. A state-feedback control scheme that is robust with respect to uncertain plant dynamics and with respect to bounded network delays, and that tolerates actuator failures, is described in [232]. The approach is a blend between passive and active FTC, since the initial controller is designed to be robust against plant model uncertainties as well as arbitrary actuator failures, and performance improvement in the case of actuator failures is achieved by means of control reconfiguration. A control adjustment technique to account for time delay variation in network control environments has been devised in [214], where the upper bound on the network delay changes abruptly due to network malfunction.

Recently, the *fault-hiding principle* has been developed for linear systems [202]. In the case of actuator faults, the reconfiguration block leads to a generalisation of the dual observer [112] called the virtual actuator [115, 123–125]. The sensor fault case leads to an observer-like solution called the virtual sensor [202]. These linear approaches have recently been extended and generalised towards new synthesis methods in [119, 171, 173, 175, 179, 180]. Experimental applications to a nonlinear thermofluid benchmark process are reported in [177, 178]. The fault-hiding principle is extended towards nonlinear dynamical systems in this monograph.

The previous discussion has focussed on methods for continuous-variable systems. At a higher level of abstraction, many processes especially in manufacturing engineering are adequately described by discrete-event dynamical systems (DEDS), short discrete-event systems. *Fault-tolerant control of discrete-event systems* (DES) has been studied in several application domains for different reasons. In communication and computer engineering, discrete-event models are used to described scheduling problems. Fault-tolerant scheduling techniques are described, for example, in [155, 230]. In automated manufacturing systems, the production process is often represented by discrete-event models, and fault-tolerant methods for improving the dependability of the production process have been suggested, with the main focus

on the computing hardware and control software [8]. Exploiting fault-tolerance of the control scheme, regardless of the software and hardware used for its implementation, is recently being studied [38, 42]. Dependable control of DES independent of specific applications starts to develop into a field in its own right. Generalised, application-independent approaches for DES are presented in [151, 152] based on languages generated by standard automata and using the concepts of safe diagnosability and safe controllability. In these works, given specification languages are recovered by means of supervisor reconfiguration. Similar automaton models are used with modified observability notions in [39], where supervisor reconfiguration is likewise proposed. Petri nets are used in [234], focussing on the satisfaction of mutual exclusion specifications.

## *Applications*

Reconfigurable control was largely motivated from applications in *flight control*, for which a large number of approaches have been developed and several tests are reported, mostly based on simulations [2, 6, 25, 28, 63]. In flight dynamics, the state variables are tightly cross-coupled. Therefore, numerous analytical redundancies arise [205], which makes the development of fault-tolerant controllers more promising than in areas such as *automotive control* and *process control*, where redundant components tend to be removed from the system design to reduce system cost. Nevertheless, applications of fault-tolerant controllers have also been reported in these areas [3, 47, 88, 144], see also the recent application-oriented book [145]. Fault-tolerant control has also been experimentally studied on *autonomous underwater vehicles* based on physical redundancies and pseudoinverse reconfiguration schemes [239]. Fault-tolerance is also important if large numbers of *railway vehicles* operate autonomously on a shared fixed track, such as it is the case in the RailCab project [66]. If a single vehicle ceases operation, it blocks the entire track. This problem also occurs in classical railway systems, however the number of vehicles on the track is considerably smaller than on the envisioned new systems.

In summary, numerous fault-tolerant and reconfigurable control approaches have been developed and studied for linear systems, but the literature on nonlinear approaches is comparatively sparse. In particular, the fault-hiding approach is not available for nonlinear systems. This monograph addresses the extension of the fault-hiding principle to two classes of nonlinear systems, namely to Hammerstein-Wiener systems and piecewise affine systems.

# Chapter 2
# Preliminaries

**Abstract.** This chapter defines the notation for this monograph and recalls central notions from the literature that constitute the theoretical foundation of this monograph. These notions are linear matrix inequalities, polyhedra and polytopes, and classical as well as recent results from stability theory. The discussion of stability theory concerns stability in the sense of Lyapunov, extensions for systems with inputs, the convergence property, and absolute stability.

## 2.1 Notation

Lower case bold letters ($\boldsymbol{x}$) denote vectors, capital bold letters ($\boldsymbol{A}$) denote matrices, and script capitals ($\mathcal{L}$) denote spaces. Systems are denoted by $\Sigma_1, \Sigma_2, \ldots$, where the subscripts distinguish different systems. The interconnection of two systems through common input/output variables is denoted by $(\Sigma_1, \Sigma_2)$. Corresponding dynamical operators are denoted by $\Omega_1, \Omega_2$ with the same distinction. The restriction of a system operator $\Omega_P$ with multiple outputs to a specific output $\boldsymbol{y}$ is denoted by $\Omega_P^{\boldsymbol{y}}$. The symbol $\triangleq$ means equal by definition.

$\mathbb{R}$ denotes the set of real numbers, and $\mathbb{R}_+ \triangleq [0, \infty)$. $\mathbb{C}$ denotes the complex plane, and $\mathbb{C}_- \triangleq \{s \in \mathbb{C} | \operatorname{Re}(s) < 0\}$. The $p$-norm of a vector $\boldsymbol{x} = (x_1, \ldots, x_n)^T$ is defined by $\|\boldsymbol{x}\|_p \triangleq \left(\sum_{i=1}^n x_i^p\right)^{1/p}$. By definition, the notation $\|\cdot\|$ refers to the vector 2-norm. For $1 \leq p \leq \infty$, and for a measurable signal $\boldsymbol{x}(t) : \mathbb{R}_+ \to \mathbb{R}^n$, the notation $\boldsymbol{x}(t) \in \mathcal{L}_p(\mathbb{R}_+, \mathbb{R}^n)$ means that $\|\boldsymbol{x}(t)\|_{\mathcal{L}_p} < \infty$, where $\|\boldsymbol{x}(t)\|_{\mathcal{L}_p} \triangleq \left(\int_{\mathbb{R}_+} \|\boldsymbol{x}(t)\|^p \mathrm{d}t\right)^{1/p}$ for $1 \leq p < \infty$ and $\|\boldsymbol{x}(t)\|_{\mathcal{L}_\infty} \triangleq \operatorname{ess\,sup}_{0 \leq \tau \leq t} \|\boldsymbol{x}(\tau)\|$. The space of locally integrable signals is denoted by $\mathcal{L}_1^{loc}$. The time argument is omitted if it is clear from the context that a signal is meant. The space of piecewise continuous signals with $m$ components is denoted by $\overline{\mathcal{PC}}_m$. The notation $\boldsymbol{x} \equiv \boldsymbol{0}$ means $\forall t : \ \boldsymbol{x}(t) = \boldsymbol{0}$.

A function $\boldsymbol{y} = \boldsymbol{\varphi}(\boldsymbol{u})$ for $\boldsymbol{y} \in \mathbb{R}^m$, $\boldsymbol{u} \in \mathbb{R}^m$ is called decomposed if $\boldsymbol{\varphi}(\boldsymbol{u}) = (\varphi_1(u_1), \ldots, \varphi_m(u_m))^T$ holds. A decomposed function $\boldsymbol{\varphi}$ is called sector-bounded in the sector $[\boldsymbol{0}, \boldsymbol{k}]$ if $\forall \boldsymbol{u} : \boldsymbol{0} \leq \frac{\boldsymbol{\varphi}(\boldsymbol{u})}{\boldsymbol{u}} \leq \boldsymbol{k}$, where $\leq$ and division are applied element-wise, which is concisely denoted by $\boldsymbol{\varphi} \in [\boldsymbol{0}, \boldsymbol{k}]$ [97]. The decomposed saturation function is defined as $\boldsymbol{u}_s = \operatorname{sat}(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u})$ where, for $i = 1, \ldots, m$,

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 23–32.
springerlink.com

$$u_{s,i} \triangleq \begin{cases} \underline{u}_i & \text{if } u_i \leq \underline{u}_i \\ u_i & \text{if } \underline{u}_i \leq u_i \leq \overline{u}_i \\ \overline{u}_i & \text{if } \overline{u}_i \leq u_i. \end{cases} \tag{2.1}$$

The following properties of matrices are used [19]. The right inverse of the matrix $\boldsymbol{A}$ is denoted by † and defined as $\boldsymbol{A}^{\dagger} \triangleq \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}$. The left inverse of the matrix $\boldsymbol{A}$ is denoted by ‡ and defined as $\boldsymbol{A}^{\ddagger} \triangleq (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$. The pseudoinverse of a matrix $\boldsymbol{A}$ satisfying all four Moore-Penrose conditions is denoted by $\boldsymbol{A}^+$ [16]. The set of eigenvalues of a matrix $\boldsymbol{A}$ is denoted by $\sigma(\boldsymbol{A})$. A matrix is called Hurwitz if all its eigenvalues have strictly negative real part. The transpose (conjugate transpose) of a real (complex) matrix $\boldsymbol{A}$ is denoted by $\boldsymbol{A}^T$ ($\boldsymbol{A}^*$). A Hermitian matrix satisfies the relation $\boldsymbol{A}^* = \boldsymbol{A}$. The notation $\boldsymbol{A} \prec 0$ ($\boldsymbol{A} \preceq 0$) for a Hermitian matrix means that the matrix $\boldsymbol{A}$ is negative (semi-) definite, meaning that all its eigenvalues are strictly negative (non-positive). Correspondingly, the symbol $\succ$ ($\succeq$) denotes positive (semi-) definiteness of Hermitian matrices. A symmetric block matrix is denoted in abbreviated form using the symbol $\star$, therefore the notation

$$\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \star & \boldsymbol{C} \end{pmatrix} \text{ abbreviates } \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{B}^T & \boldsymbol{C} \end{pmatrix}.$$

A pair of matrices $(\boldsymbol{A}, \boldsymbol{B})$, where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$, is called stabilisable, if there exists a matrix $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ such that $\boldsymbol{A} - \boldsymbol{B}\boldsymbol{K}$ is Hurwitz. A pair of matrices $(\boldsymbol{C}, \boldsymbol{A})$, where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{C} \in \mathbb{R}^{q\times n}$, is called detectable, if there exists a matrix $\boldsymbol{L} \in \mathbb{R}^{n\times q}$ such that $\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}$ is Hurwitz.

The following comparison functions are used [104]. A function $F : S \to \mathbb{R}$ defined on a set $S \subset \mathbb{R}^n$ containing zero is *positive definite* if $F(\boldsymbol{x}) > 0$ holds for all $\boldsymbol{x} \in S, \boldsymbol{x} \neq \boldsymbol{0}$, and $F(\boldsymbol{0}) = 0$. A *class* $\mathcal{K}$ *function* is a function $\alpha : \mathbb{R}_+ \to \mathbb{R}_+$ which is continuous, strictly increasing, and satisfies $\alpha(0) = 0$. Any function $\alpha$ that satisfies these requirements is said to be in the class $\mathcal{K}$, denoted by $\alpha \in \mathcal{K}$. A *class* $\mathcal{K}_\infty$ *function* is a function $\alpha \in \mathcal{K}$ that is additionally unbounded, i.e. $\lim_{s\to\infty} \alpha(s) = \infty$. A *class* $\mathcal{KL}$ *function* is a function $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \to \mathbb{R}_+$ such that $\beta(\cdot, t) \in \mathcal{K}$ for any fixed $t$, and for each fixed $r \geq 0$, $\beta(r, t) \to 0$ as $t \to \infty$. The unit step function $\rho(t)$ is defined for $t \in \mathbb{R}$ as $\rho(t) = 0$ if $t < 0$ and $\rho(t) = 1$ if $t \geq 0$. The space of $k$ times continuously differentiable functions is denoted by $C^k$.

Throughout the monograph, the end of a definition is marked by means of a diamond ($\diamond$), the end of a remark is marked by means of a circle ($\circ$), and the end of proofs placed in the main parts of the monograph is marked by means of black squares ($\blacksquare$).

## 2.2 Linear Matrix Inequalities

Numerous standard control problems can be readily formulated as (nonlinear) matrix inequalities [29, 187]. Their transformation into a linear form can often be achieved by means of standard transformations.

**Definition 2.1 (Linear matrix inequalities [29]).** Let $\boldsymbol{F}_i = \boldsymbol{F}_i^T \in \mathbb{R}^{m\times m}$ be a family of symmetric parameter matrices.

- **Linear matrix inequality:** The inequality

$$\sum_{i=1}^{m} x_i \boldsymbol{F}_i \succ \boldsymbol{0} \tag{2.2}$$

  is a *linear matrix inequality* (LMI), where $\boldsymbol{x} = (x_1, \ldots, x_m)^T \in \mathbb{R}^m$ is called the decision variable.
- **Congruence transformation:** Given a Hermitian matrix $\boldsymbol{A}$ and a square nonsingular matrix $\boldsymbol{T}$, the transformation

$$\boldsymbol{A} \to \boldsymbol{T}^* \boldsymbol{A} \boldsymbol{T} \tag{2.3}$$

  is called a *congruence transformation* of $\boldsymbol{A}$.
- **Inertia:** Let $\nu_-(\boldsymbol{A})$, $\nu_0(\boldsymbol{A})$, and $\nu_+(\boldsymbol{A})$ denote the numbers (counting multiplicity) of eigenvalues of a symmetric matrix $\boldsymbol{A}$ with negative, zero, and positive real part, respectively. The triple $\mathrm{In}(\boldsymbol{A}) \triangleq (\nu_-(\boldsymbol{A}), \nu_0(\boldsymbol{A}), \nu_+(\boldsymbol{A}))^T$ is called the *inertia* of $\boldsymbol{A}$. ◇

The following lemma identifies congruence transformations as inertia-preserving operations.

**Lemma 2.1 (Congruence transformation [29]).** *If $\boldsymbol{A}$ is Hermitian and $\boldsymbol{T}$ is nonsingular, then the matrices $\boldsymbol{A}$ and $\boldsymbol{T}^*\boldsymbol{A}\boldsymbol{T}$ have the same inertia:* $\mathrm{In}(\boldsymbol{A}) = \mathrm{In}(\boldsymbol{T}^*\boldsymbol{A}\boldsymbol{T})$.

As a consequence of Lemma 2.1, congruence transformations are equivalence operations on LMIs in the sense that $\boldsymbol{A} \succ 0$ if and only if $\boldsymbol{T}^*\boldsymbol{A}\boldsymbol{T} \succ 0$. Together with the following Lemma, congruence transformations are often the key to transforming nonlinear matrix inequalities into equivalent linear matrix inequalities.

**Lemma 2.2 (Schur complements [29]).** *The following statements are equivalent:*

1. $\begin{pmatrix} \boldsymbol{Q} & \boldsymbol{S} \\ \boldsymbol{S}^T & \boldsymbol{R} \end{pmatrix} \prec 0$
2. $\boldsymbol{Q} \prec 0$ *and* $\boldsymbol{R} - \boldsymbol{S}^T\boldsymbol{Q}^{-1}\boldsymbol{S} \prec 0$
3. $\boldsymbol{R} \prec 0$ *and* $\boldsymbol{Q} - \boldsymbol{S}\boldsymbol{R}^{-1}\boldsymbol{S}^T \prec 0$.

The equivalence relations defined by Schur complements also hold for reversed inequalities ($\succ$ instead of $\prec$).

**Definition 2.2 (Affine combination and affine independence [30]).** Given is a set of vectors $\mathcal{X} \triangleq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}$, which is used to define the following two notions.

- **Affine combination:** A point

$$\boldsymbol{p} = \sum_{i=1}^{k} \alpha_i \boldsymbol{x}_i \text{ where } \sum_{i=1}^{k} \alpha_i = 1 \tag{2.4}$$

  is called an *affine combination* of the vectors in $\mathcal{X}$.

- **Affine independence:** The vectors in the set $\mathcal{X}$ are *affinely independent*, if all vectors in the set

$$\{\boldsymbol{x}_2 - \boldsymbol{x}_1, \ldots, \boldsymbol{x}_k - \boldsymbol{x}_1\} \tag{2.5}$$

  are linearly independent. ◇

LMIs are nowadays efficiently solvable using numerical methods, for example interior point methods [30], implemented in software such as YALMIP and Sedumi [109, 207].

## 2.3 Polyhedra and Polytopes

A polyhedron $\Lambda$ is a (not necessarily bounded) set defined by a finite number of linear inequalities, whereas a polytope is always bounded [253]. Polyhedra and polytopes can be defined as intersections of finite numbers of half-spaces as follows.

**Definition 2.3 (Polyhedra and polytopes [253])**

- **Polyhedron:** A *polyhedron* $\Lambda \subseteq \mathbb{R}^n$ is a set

$$\Lambda = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{H}\boldsymbol{x} \leq \boldsymbol{k}\}, \tag{2.6}$$

  where $\boldsymbol{H} \in \mathbb{R}^{n \times n}$ is a matrix where every row is a normal direction to one of the hyperplanes, and $\boldsymbol{k} \in \mathbb{R}^{n \times 1}$ gives the hyperplane offsets.
- **Polytope:** A *polytope* $\mathcal{P} \subseteq \mathbb{R}^n$ is a bounded and closed set

$$\mathcal{P} = \{\boldsymbol{x} \in \mathbb{R}^n : \boldsymbol{H}\boldsymbol{x} \leq \boldsymbol{k}\}, \tag{2.7}$$

  where the matrix $\boldsymbol{H} \in \mathbb{R}^{n \times n}$ defines the normal directions to the hyperplane, and $\boldsymbol{k} \in \mathbb{R}^{n \times 1}$ defines the hyperplane offsets. ◇

A polytope $\mathcal{P}$ is thus a compact polyhedron. Every polytope is a polyhedron, but not vice versa. Furthermore, every polyhedron and every polytope is convex, which is immediate from their definitions.

The representation (2.7) is often called $(\boldsymbol{H}/\boldsymbol{k})$-representation in the literature. The interior of any set (in particular, of a polytope) is denoted by $\text{int}(\mathcal{P})$. Alternatively, a polytope is defined by the convex hull of a finite set of vertex points.

**Definition 2.4 (Convex hull [253]).** The convex hull $\text{co}(\mathcal{X})$ of a set of given points $\mathcal{X} \triangleq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k\}$ is defined by

$$\text{co}(\mathcal{X}) = \left\{ \sum_{i=1}^{k} \alpha_i \boldsymbol{x}_i : \boldsymbol{x}_i \in \mathcal{X}, \alpha_i \in \mathbb{R}_+, \sum_{i=1}^{k} \alpha_i = 1 \right\}. \tag{2.8}$$

The set $\text{co}(\mathcal{X})$ is a polytope with the set of vertex points $\mathcal{V} \subseteq \mathcal{X}$. ◇

In the course of this monograph, the class of simplex polytopes will be of special interest.

**Definition 2.5 (Simplices and Delaunay partitions [244, 253])**

- **Simplex:** A *simplex* $\mathcal{S}$ in $\mathbb{R}^n$ is the convex hull of $n+1$ affinely independent points $\mathcal{X} \triangleq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n+1}\}$:

$$\mathcal{S} = \mathrm{co}(\mathcal{X}). \tag{2.9}$$

  The points $\boldsymbol{x}_i \in \mathcal{X}$, $i \in \{1, \ldots, n+1\}$ are called *vertices*, and the convex hull of any $n$ vertices is called a *facet*.
- **Delaunay partition:** Let $\mathcal{V} \subseteq \mathbb{R}^n$ be a compact subset of the state-space and $\mathcal{S}_i$, $i \in I = \{1, \ldots, k\}$ be a collection of nonintersecting simplices that satisfy the conditions

$$\bigcup_{i=1}^{k} \mathcal{S}_i = \mathcal{V}$$
$$\forall i, j \in I, i \neq j : \mathrm{int}\mathcal{S}_i \cap \mathrm{int}\mathcal{S}_j = \emptyset.$$

  Then, the set of simplices

$$\mathcal{D} \triangleq \{\mathcal{S}_1, \ldots, \mathcal{S}_k\}$$

  is called a *Delaunay partition* of the set $\mathcal{V}$. ◇

Simplices and Delaunay partitions are used in connection with piecewise affine systems in this monograph.

## 2.4 Stability Theory

### Lyapunov-Type Stability of Equilibria

The following stability definitions are defined for general nonlinear systems of the form

$$\Sigma : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t)) \end{cases} \qquad \boldsymbol{x}(0) = \boldsymbol{x}_0, \tag{2.10}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state, $\boldsymbol{u}(t) \in \mathbb{R}^m$ is an input, $\boldsymbol{y}(t) \in \mathbb{R}^q$ is the output, and $\boldsymbol{f}(\cdot,\cdot)$, $\boldsymbol{h}(\cdot)$ are nonlinear functions of appropriate dimensions. First, stability notions for the autonomous system (2.10) are defined ($\boldsymbol{u} \equiv \boldsymbol{0}$).

**Definition 2.6 (0-global asymptotic stability and 0-global exponential stability [97]).** The system (2.10) with zero input $\boldsymbol{u} \equiv \mathbf{0}$ is called

- *0-globally asymptotically stable* (0-GAS), if for all $t_0$ there exists a class $\mathcal{KL}$ function $\beta$ such that for all initial conditions $\boldsymbol{x}(t_0)$

$$\|\boldsymbol{x}(t)\| \leq \beta(\|\boldsymbol{x}(t_0)\|, t - t_0) \; \forall t \geq t_0.$$

  If the function $\beta$ does not depend on $t_0$, then the system (2.10) with zero input $\boldsymbol{u} \equiv \mathbf{0}$ is called *0-globally uniformly asymptotically stable*.
- *0-globally exponentially stable* (0-GES), if for all initial times $t_0$ and all corresponding initial conditions $\boldsymbol{x}(t_0)$, the corresponding solution $\boldsymbol{x}$ satisfies

$$\|\boldsymbol{x}(t)\| \leq c\|\boldsymbol{x}(t_0)\|\mathrm{e}^{-\lambda(t-t_0)} \; \forall t \geq t_0$$

  for some real numbers $c, \lambda > 0$. ◇

In other words, exponential stability is a special case of asymptotic stability where the function $\beta$ is exponential in the time argument.

**Definition 2.7 (0-globally asymptotically stable solutions [159]).** A solution $\bar{\boldsymbol{x}}(t)$ of the system (2.10) with zero input $\boldsymbol{u} \equiv \mathbf{0}$ starting at the initial condition $\bar{\boldsymbol{x}}(t_0)$ is called *0-globally asymptotically stable*, if for all initial times $t_0$ there exists a class $\mathcal{KL}$ function $\beta$ such that for all initial conditions $\boldsymbol{x}(t_0)$, the corresponding solution $\boldsymbol{x}(t)$ satisfies

$$\|\bar{\boldsymbol{x}}(t) - \boldsymbol{x}(t)\| \leq \beta(\|\bar{\boldsymbol{x}}(t_0) - \boldsymbol{x}(t_0)\|, t - t_0) \; \forall t \geq t_0.$$

If the function $\beta$ does not depend on $t_0$, then the solution $\bar{\boldsymbol{x}}(t)$ is called *0-globally uniformly asymptotically stable*. ◇

**Theorem 2.1 (Converse Lyapunov theorem [97]).** *Let $\boldsymbol{x} = \mathbf{0}$ be an equilibrium point for the nonlinear time-invariant system $\dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t), \mathbf{0})$ with zero input, where $\boldsymbol{f}(\cdot,\cdot)$ is defined in Equation (2.10). If the autonomous nonlinear system is globally exponentially stable, then there exists a continuously differentiable function $V : \mathbb{R}^n \to \mathbb{R}$ that satisfies the inequalities*

$$c_1\|\boldsymbol{x}\|^2 \leq V(\boldsymbol{x}) \leq c_2\|\boldsymbol{x}\|^2$$
$$\nabla V(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}, \mathbf{0}) \leq -c_3\|\boldsymbol{x}\|^2, \quad \|\nabla V(\boldsymbol{x})\| \leq c_4\|\boldsymbol{x}\|,$$

*where $c_1, c_2, c_3, c_4 > 0$.*

For systems with inputs, the notions of input-to-state stability (ISS), input-to-state practical stability (ISpS), and input-to-output-stability (IOS) are useful to characterise the boundedness of solutions $\boldsymbol{x}(t)$ of the system (2.10) in the presence of inputs.

**Definition 2.8 (Input-to-state stability, input-to-state practical stability, input-to-output stability [89, 198]).** The system (2.10) is called

- globally *input-to-state stable* (ISS) with respect to (w.r.t.) the input $\boldsymbol{u}$, if

$$\exists \beta \in \mathcal{KL},\ \gamma \in \mathcal{K}:\ \|\boldsymbol{x}(t)\| \leq \beta(\|\boldsymbol{x}(t_0)\|, t - t_0) + \gamma(\|\boldsymbol{u}\|_{\mathcal{L}_\infty}) \tag{2.11}$$

  for all inputs $\boldsymbol{u}$, all initial conditions $\boldsymbol{x}(t_0)$, and all times $t \geq t_0$, where $\boldsymbol{x}$ is the solution of (2.10),
- globally *input-to-state practically stable* (ISpS) w.r.t. the input $\boldsymbol{u}$, if

$$\exists \beta \in \mathcal{KL},\ \gamma \in \mathcal{K},\ K > 0:\ \|\boldsymbol{x}(t)\| \leq \beta(\|\boldsymbol{x}(t_0)\|, t - t_0) + \gamma(\|\boldsymbol{u}\|_{\mathcal{L}_\infty}) + K \tag{2.12}$$

  for all inputs $\boldsymbol{u}$, all initial conditions $\boldsymbol{x}(t_0)$, and all times $t \geq t_0$, where $\boldsymbol{x}$ is the solution of (2.10), and
- globally *input-to-output stable* (IOS) w.r.t. the input $\boldsymbol{u}$ and the output $\boldsymbol{y}$, if

$$\exists \beta \in \mathcal{KL},\ \gamma \in \mathcal{K}:\ \|\boldsymbol{y}(t)\| \leq \beta(\|\boldsymbol{x}(t_0)\|, t - t_0) + \gamma(\|\boldsymbol{u}\|_{\mathcal{L}_\infty}) \tag{2.13}$$

  for all inputs $\boldsymbol{u}$, all initial conditions $\boldsymbol{x}(t_0)$, and all times $t \geq t_0$, where $\boldsymbol{y}$ is the output of the system (2.10). ◇

The state of an ISS system is bounded for arbitrary bounded control inputs, and the unforced part of the solution asymptotically vanishes. Thus for zero inputs, the definition of ISS recovers the 0-GAS property. ISS is understood as a global property throughout this monograph. The ISS property requires that the system asymptotically converges to the origin in the absence of inputs. The weaker notion of ISpS only requires convergence to a neighbourhood of the origin. When dealing with systems interconnected through inputs and outputs, the notion of IOS systems is more useful than the notion of ISS.

The following theorem links the ISS notion to the concept of Lyapunov functions.

**Theorem 2.2 (Lyapunov criterion for ISS systems [198]).** *Let $V : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function such that*

$$\alpha_1(\|\boldsymbol{x}\|) \leq V(\boldsymbol{x}) \leq \alpha_2(\|\boldsymbol{x}\|) \tag{2.14}$$

$$\nabla V(\boldsymbol{x}) \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \leq -W(\boldsymbol{x}),\ \forall \|\boldsymbol{x}\| \geq \rho(\|\boldsymbol{u}\|) > 0 \tag{2.15}$$

*for all $(\boldsymbol{x}, \boldsymbol{u}) \in \mathbb{R}^n \times \mathbb{R}^m$, where $\alpha_1$, $\alpha_2 \in \mathcal{K}_\infty$, $\rho \in \mathcal{K}$, and $W$ continuous and positive definite on $\mathbb{R}^n$. Then, the nonlinear system (2.10) is ISS with $\gamma = \alpha_1^{-1} \circ \alpha_2 \circ \rho$.*

**Property 2.1 (ISS of cascaded ISS systems [198])**

$$\textit{Let the system } \begin{cases} \dot{\boldsymbol{v}}(t) &= \boldsymbol{f}(\boldsymbol{v}(t), \boldsymbol{w}(t), \boldsymbol{u}(t)), \quad \boldsymbol{v}(t) \in \mathbb{R}^s \\ \dot{\boldsymbol{w}}(t) &= \boldsymbol{g}(\boldsymbol{w}(t), \boldsymbol{u}(t)), \quad \boldsymbol{w}(t) \in \mathbb{R}^n \end{cases} \tag{2.16}$$

*be such that the $\boldsymbol{v}$-subsystem is globally ISS w.r.t. the input $(\boldsymbol{w}, \boldsymbol{u})$, and the $\boldsymbol{w}$-subsystem is globally ISS w.r.t. the input $\boldsymbol{u}$. Then, the cascade system (2.16) is globally ISS w.r.t. the input $\boldsymbol{u}$.*

**Property 2.2 (IOS of cascaded IOS systems [89]).** *Let the system*

$$\begin{cases} \dot{\boldsymbol{v}}(t) &= \boldsymbol{f}(\boldsymbol{v}(t), \boldsymbol{p}(t), \boldsymbol{u}(t)), \quad \boldsymbol{v}(t) \in \mathbb{R}^s \\ \dot{\boldsymbol{w}}(t) &= \boldsymbol{g}(\boldsymbol{w}(t), \boldsymbol{u}(t)), \quad \boldsymbol{w}(t) \in \mathbb{R}^n \\ \boldsymbol{p}(t) &= \boldsymbol{h}_1(\boldsymbol{w}(t), \boldsymbol{u}(t)) \\ \boldsymbol{q}(t) &= \boldsymbol{h}_2(\boldsymbol{v}(t), \boldsymbol{p}(t), \boldsymbol{u}(t)) \end{cases} \tag{2.17}$$

*be such that the $\boldsymbol{v}$-subsystem with the input $(\boldsymbol{p}, \boldsymbol{u})$ and output $\boldsymbol{q}$ is IOS, and the $\boldsymbol{w}$-subsystem is IOS w.r.t. the input $\boldsymbol{u}$ and the output $\boldsymbol{p}$. Then, the cascade connection (2.16) is IOS w.r.t. the input $\boldsymbol{u}$ and the output $(\boldsymbol{p}, \boldsymbol{q})$.*

Similarly, the concepts of finite-gain stability and the $H_\infty$-system norm of a given system are defined.

**Definition 2.9 (Finite-gain $\mathcal{L}$ stability, $\mathcal{L}_2$-gain, $H_\infty$-system norm [200]).** The system $\Sigma$ defined in Equation (2.10) with $\boldsymbol{x}_0 = \boldsymbol{0}$ is called

- *finite-gain $\mathcal{L}$ stable* if there exist scalars $\gamma \geq 0$ and $K \geq 0$ such that

$$\|\boldsymbol{y}(t)\|_{\mathcal{L}_2} \leq \gamma \|\boldsymbol{u}(t)\|_{\mathcal{L}_2} + K \tag{2.18}$$

  holds for all input/output pairs $(\boldsymbol{u}, \boldsymbol{y})$ of $\Sigma$, and
- the smallest number $\gamma > 0$ that satisfies Inequality (2.18) for all input/output pairs $(\boldsymbol{u}, \boldsymbol{y})$ is called the *$\mathcal{L}_2$-gain* or, if the system (2.10) is linear, the *$H_\infty$-norm* of the system $\Sigma$, denoted by $\|\Sigma\|_{H_\infty}$. ◇

The following theorem is useful for studying the stability properties of feedback interconnections of stable systems.

**Theorem 2.3 (Small-gain theorem [97]).** *Let $\Sigma_1$ and $\Sigma_2$ be finite-gain $\mathcal{L}$ stable systems with respect to the input/output signals $(\boldsymbol{u}_1, \boldsymbol{y}_1)$ and $(\boldsymbol{u}_2, \boldsymbol{y}_2)$ and the $\mathcal{L}_2$-gains $\gamma_1$ and $\gamma_2$. Then, the feedback interconnection $(\boldsymbol{u}_1 = \boldsymbol{y}_2 + \boldsymbol{r}_1)$, $(\boldsymbol{u}_2 = \boldsymbol{y}_1 + \boldsymbol{r}_2)$ is finite-gain stable if $\gamma_1 \gamma_2 < 1$.*

### Lyapunov-Type Stability of Non-equilibrium Solutions

The following notion describes the stability of all possible solutions of the system (2.10) with respect to each other, for a given input $\boldsymbol{u}$ but starting from different initial conditions.

**Definition 2.10 (Incremental global asymptotic stability [5]).** The system (2.10) is called *incrementally globally asymptotically stable* ($\delta$GAS), if there exists a function $\beta \in \mathcal{KL}$ such that for all $\boldsymbol{u}$, and all pairs $(\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t_0), \boldsymbol{x}_{\boldsymbol{u}}(t_0))$ the relation

$$\|\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t) - \boldsymbol{x}_{\boldsymbol{u}}(t)\| \leq \beta(\|\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t_0) - \boldsymbol{x}_{\boldsymbol{u}}(t_0)\|, t - t_0) \; \forall t \geq t_0$$

is satisfied. ◇

Sometimes, only the stability of a single solution is of interest.

**Definition 2.11 (Globally asymptotically stable solutions [159]).** A solution $\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t)$ of the system (2.10) for the input $\boldsymbol{u}$ is called *globally asymptotically stable*, if for all initial times $t_0$ there exists a function $\beta \in \mathcal{KL}$ such that every solution $\boldsymbol{x}_{\boldsymbol{u}}(t)$ starting from an arbitrary initial condition $\boldsymbol{x}_{\boldsymbol{u}}(t_0)$ satisfies the relation

$$\|\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t) - \boldsymbol{x}_{\boldsymbol{u}}(t)\| \leq \beta(\|\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t_0) - \boldsymbol{x}_{\boldsymbol{u}}(t_0)\|, t - t_0) \; \forall t \geq t_0.$$

If the function $\beta$ does not depend on $t_0$, then the solution $\bar{\boldsymbol{x}}_{\boldsymbol{u}}(t)$ is called *globally uniformly asymptotically stable*. ◇

The following property is stronger than the stability of solutions and will be instrumental in obtaining the tracking properties for piecewise affine systems.

**Definition 2.12 (Convergence [45, 159]).** The system (2.10) with the piecewise-continuous input $\boldsymbol{u} \in \overline{\mathcal{PC}_m}$ is said to be

- *convergent* if for all inputs $\boldsymbol{u} \in \overline{\mathcal{PC}_m}$ there exists a solution $\bar{\boldsymbol{x}}_u$ satisfying the following conditions
  1. $\bar{\boldsymbol{x}}_u(t)$ is defined and bounded for all $t \in \mathbb{R}$,
  2. $\bar{\boldsymbol{x}}_u(t)$ is globally asymptotically stable.
- *uniformly convergent* if it is convergent and, for all inputs $\boldsymbol{u} \in \overline{\mathcal{PC}_m}$, $\bar{\boldsymbol{x}}_u$ is globally uniformly asymptotically stable.
- *exponentially convergent* if it is convergent and, for all inputs $\boldsymbol{u} \in \overline{\mathcal{PC}_m}$, $\bar{\boldsymbol{x}}_u$ is globally exponentially stable. ◇

Uniform convergence implies that the steady-state solution $\bar{\boldsymbol{x}}_u$ of the system (2.10) is unique[1] and depends only on the input signal $\boldsymbol{u}$. In other words, a uniformly convergent system "forgets" its initial condition. If the input $\boldsymbol{u}$ to a uniformly convergent system is periodic with the period $T$, then its steady-state solution $\bar{\boldsymbol{x}}_{\boldsymbol{u}}$ is periodic with the period $T$ [159]. In particular, if the input is constant, then the steady-state solution is constant.

**Theorem 2.4 (Sufficient condition for convergence [159]).** *Consider the system (2.10) and let $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})$ be continuous with respect to $\boldsymbol{u} \in \mathbb{R}^m$ and locally Lipschitz with respect to $\boldsymbol{x} \in \mathbb{R}^n$. Moreover, let $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})$ be $C^1$ with respect to $\boldsymbol{x}$ in $(\boldsymbol{x},\boldsymbol{u}) \in (\mathbb{R}^n \backslash \Gamma) \times \mathbb{R}^m$, where $\Gamma \subset \mathbb{R}^n$ is a set consisting of a finite number of hyperplanes given by equations of the form $\boldsymbol{h}_j^T \boldsymbol{x} + k_j = 0$, for some $\boldsymbol{h}_j \in \mathbb{R}^n$ and $k_j \in \mathbb{R}$, $j = 1, \ldots, k$. Suppose that there exist matrices $\boldsymbol{P} = \boldsymbol{P}^T \succ 0$ and $\boldsymbol{Q} = \boldsymbol{Q}^T \succ 0$ such that*

$$\boldsymbol{P}\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x},\boldsymbol{u}) + \frac{\partial \boldsymbol{f}^T}{\partial \boldsymbol{x}}(\boldsymbol{x},\boldsymbol{u})\boldsymbol{P} \preceq -\boldsymbol{Q}, \quad \forall \boldsymbol{x} \in \mathbb{R}^n \backslash \Gamma, \, \boldsymbol{u} \in \mathbb{R}^m. \tag{2.19}$$

[1] Note that while $\delta$GAS is closely related to convergence, a $\delta$GAS system does not necessarily have a unique steady-state solution. Rather, all steady-state solutions converge to each other.

*Then, the system (2.10) is globally exponentially convergent for the class of inputs* $\boldsymbol{u} \in \overline{\mathcal{PC}}_m$.

The inequality (2.19) is known as the Demidovic condition. For linear systems, it reduces to the Lyapunov inequality.

### Absolute Stability

Consider the following multi-input/multi-output strictly proper linear dynamical systems with the same number of inputs and outputs in closed loop with a nonlinear function $\boldsymbol{\varphi}(\cdot)$ in the feedback branch,

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t), \quad \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{u}(t) &= -\boldsymbol{\varphi}(\boldsymbol{y}(t)), \end{cases} \tag{2.20}$$

where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$, $\boldsymbol{C} \in \mathbb{R}^{q\times n}$, and $q = m$, that is, the system has the same number of inputs as outputs. It is assumed throughout the monograph that the function $\boldsymbol{\varphi}(\cdot)$ is

- decomposed, namely $\boldsymbol{\varphi}(\boldsymbol{u}) = (\varphi_1(u_1), \ldots, \varphi_m(u_m))^T$, and
- sector-bounded within the first and third quadrant, namely $\boldsymbol{\varphi} \in [\boldsymbol{0}, \boldsymbol{k}]$.

The diagonal matrix $\boldsymbol{S}$ of inverse sector bounds is defined to be

$$\boldsymbol{S} = \operatorname{diag}(1/k_1, \ldots, 1/k_m). \tag{2.21}$$

**Definition 2.13 (Absolute stability [97]).** Consider the system (2.20), where the pair $(\boldsymbol{A}, \boldsymbol{B})$ is controllable, and the pair $(\boldsymbol{C}, \boldsymbol{A})$ is observable. The system (2.20) is said to be *absolutely stable* in the sector $[0, \boldsymbol{k}]$, if the origin is a globally uniformly asymptotically stable equilibrium point for all functions $\boldsymbol{\varphi} \in [0, \boldsymbol{k}]$. ⋄

The sector requirement in the above definition is strict, and the nonlinear function must obey it at every time.

**Theorem 2.5 (LMI-based criterion for absolute stability [29]).** *The system (2.20) is absolutely stable in the section* $[0, \boldsymbol{k}]$*, if there exists a solution* $\boldsymbol{X} = \boldsymbol{X}^T > 0$ *to the LMI*

$$\left(\begin{array}{c|c} -(\boldsymbol{A}^T\boldsymbol{X} + \boldsymbol{X}\boldsymbol{A}) & \boldsymbol{C}^T - \boldsymbol{X}\boldsymbol{B} \\ \hline \star & 2\boldsymbol{S} \end{array}\right) > 0. \tag{2.22}$$

The feasibility of the LMI (2.22) is equivalent to strict positive realness of the transfer function $\boldsymbol{T}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}$, and equivalent to the strict passivity of the linear subsystem of the system (2.20).

# Chapter 3
# Reconfigurable Control Problem and Fault-Hiding Approach

**Abstract.** This chapter defines the reconfiguration problem, which consists in the recovery of the nominal stability, setpoint tracking, and performance properties by the reconfigured closed-loop system. First, reconfiguration problems are formulated based on model-matching ideas. Second, the fault-hiding concept is introduced and the reconfiguration problems are formulated in this context. The fault-hiding concept is the basis for all reconfigurable control solutions presented in later chapters. The general properties of the fault-hiding approach are explained in more detail.

## 3.1 General Dynamical Operators

It will be useful to refer to dynamical systems from an input/output perspective without specifying inner details. Dynamical operators are used for this purpose. In this monograph, operators may be regarded as an abbreviated way of specifying relationships between input and output signals without writing the detailed state-space model. Nevertheless, it should be kept in mind that the internal dynamics are always realised by a state-space model throughout this monograph.

**Definition 3.1 (Dynamical operator).** A dynamical operator $\Omega_P$ is a map with memory $\Omega_P : \mathcal{L}_1^{loc}(\mathbb{R}^m) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \to \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathcal{L}_1^{loc}(\mathbb{R}^p)$ that maps input signals $\boldsymbol{u}_c(t) \in \mathcal{L}_1^{loc}(\mathbb{R}^m)$, $\boldsymbol{d}(t) \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and an initial condition $\boldsymbol{x}_0 \in \mathbb{R}^n$ to output signals $\boldsymbol{y}(t) \in \mathcal{L}_1^{loc}(\mathbb{R}^q)$, $\boldsymbol{z}(t) \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$,

$$(\boldsymbol{y}, \boldsymbol{z}) = \Omega_P(\boldsymbol{u}_c, \boldsymbol{d}, \boldsymbol{x}_0). \tag{3.1}$$

The memory is represented by an internal state variable $\boldsymbol{x}$, whose initial condition is $\boldsymbol{x}_0$. ◇

## 3.2 Nominal Nonlinear Systems

This monograph is concerned with dynamical systems with inputs and outputs that are represented by time-invariant ordinary differential equations.

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 33–54.
springerlink.com 

**Definition 3.2 (Nonlinear dynamical system [97, 186]).** A *nonlinear dynamical system* is a set of first-order ordinary differential equations (ODEs)

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t), \boldsymbol{u}_c(t), \boldsymbol{d}(t)) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{x}(t)) \\ \boldsymbol{z}(t) &= \boldsymbol{h}_z(\boldsymbol{x}(t)), \end{cases} \tag{3.2}$$

where $\boldsymbol{x}(t) \in \mathbb{R}^n$ is the state, $\boldsymbol{u}_c(t) \in \mathbb{R}^m$ is the control input, $\boldsymbol{d}(t) \in \mathbb{R}^k$ is the disturbance input, $\boldsymbol{y}(t) \in \mathbb{R}^q$ is the measured output, $\boldsymbol{z}(t) \in \mathbb{R}^p$ is the controlled output, and

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{3.3}$$

is the initial condition. ◇

The nonlinear dynamical system is shown in Fig. 3.1.

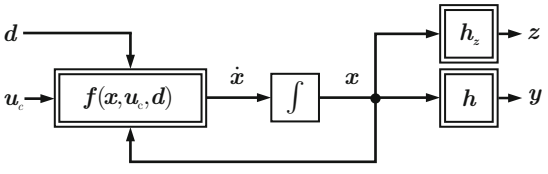


**Fig. 3.1** Nonlinear dynamical system.

The system (3.2) is called

- *continuous* if the function $\boldsymbol{f}(\cdot,\cdot,\cdot)$ is continuous in all variables,
- *smooth* if $\boldsymbol{f}(\cdot,\cdot,\cdot) \in C^\infty$,
- *linear* if all functions $\boldsymbol{f}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}_z(\cdot,\cdot,\cdot)$ are linear in their arguments,
- *nonlinear* if at least one of the functions $\boldsymbol{f}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}_z(\cdot,\cdot,\cdot)$ is not linear in its arguments.

A *classical solution* of (3.2) on the time interval $[t_0, t_1]$ is a continuously differentiable function $x : [t_0, t_1] \to \mathbb{R}^n$ that satisfies (3.2). Every continuous nonlinear system (3.2) with initial condition (3.3) admits a classical solution, which is unique if the function $\boldsymbol{f}$ is locally Lipschitz [41].

**Example 3.1 (Nonlinear model of a ship).** *The ship model defined in Equations (1.1)–(1.4) has the special form*

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{g}(\boldsymbol{x}(t)) + \boldsymbol{B}\,\mathrm{sat}(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u}(t)) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \dot{\psi}(t) &= x_3(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(\boldsymbol{x}(t)^T, \psi(t))^T \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z(\boldsymbol{x}(t)^T, \psi(t))^T, \end{cases} \tag{3.4}$$

*where* $\boldsymbol{x} = (v,w,r)^T$, $\boldsymbol{u} = (u_1,u_2,u_3)^T$, $\underline{\boldsymbol{u}} = (-1,-1,-1)^T$, $\overline{\boldsymbol{u}} = (1,1,1)^T$, $\boldsymbol{y} = (v,r,\psi)^T$, $\boldsymbol{z} = (v,\psi)^T$, *and* $\boldsymbol{d} = (a_v,a_w)^T$. *With reference to the system (3.2), the function* $\boldsymbol{f}$ *has the special form* $\boldsymbol{f}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d}) = \boldsymbol{g}(\boldsymbol{x}) + \boldsymbol{B}\,\mathrm{sat}(\underline{\boldsymbol{u}},\overline{\boldsymbol{u}},\boldsymbol{u}) + \boldsymbol{B}\boldsymbol{d}$. *The autonomous part* $\boldsymbol{g}(\boldsymbol{x}(t))$ *of the model is nonlinear, whereas the saturated control input* $\mathrm{sat}(\underline{\boldsymbol{u}},\overline{\boldsymbol{u}},\boldsymbol{u}(t))$ *and the disturbance input* $\boldsymbol{d}(t)$ *enter linearly. The output functions* $\boldsymbol{h}(\cdot)$ *and* $\boldsymbol{h}_z(\cdot)$ *are linear. The nonlinearities of the autonomous part of the system are bilinear, that is, they consist of products of state variable pairs. In order to analyse the system, the vector field defined by the function* $\boldsymbol{g}(\cdot)$ *is illustrated. Fig. 3.2 shows the directions of the vector field.*

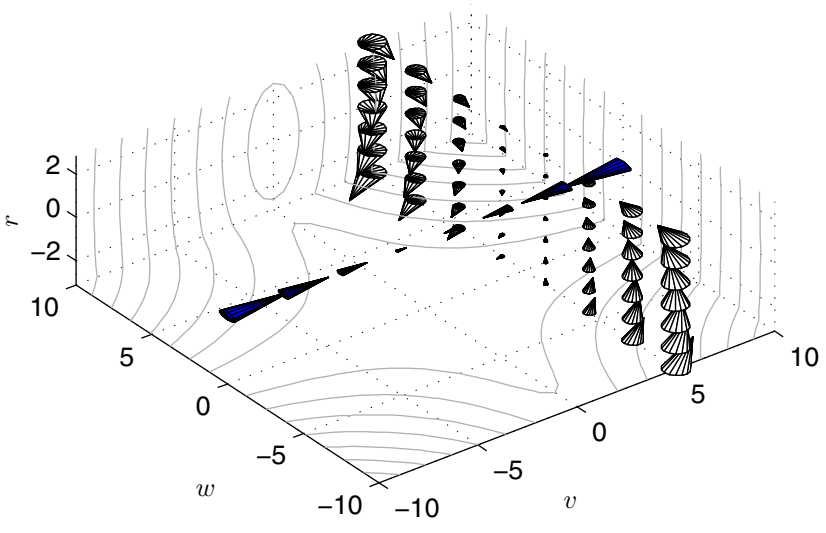


**Fig. 3.2** Illustration of ship vector field by means of flow vectors.

All reconfiguration methods presented in this monograph concern nonlinear systems. However, instead of the general form (3.2), two subclasses of nonlinear systems are used in the development of reconfigurable control strategies. These classes of nonlinear systems are Hammerstein-Wiener systems and piecewise affine systems.

## 3.3 Nominal Closed-Loop System and Assumptions

In order to formulate the reconfiguration problem, the nominal plant models stated in the previous section is complemented with a nominal controller to obtain a nominal closed-loop system. Consider the nominal controller

$$\Sigma_C : \begin{cases} \dot{\boldsymbol{x}}_c(t) &= \boldsymbol{f}_c(\boldsymbol{x}_c(t), \boldsymbol{y}(t), \boldsymbol{r}(t)) \\ \boldsymbol{u}_c(t) &= \boldsymbol{h}_c(\boldsymbol{x}_c(t), \boldsymbol{y}(t), \boldsymbol{r}(t)) \end{cases} \tag{3.5}$$
$$\boldsymbol{x}_c(0) = \boldsymbol{x}_{c0}$$

for the nominal nonlinear plant (3.2), with that is associated an operator $\Omega_C : \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathcal{L}_1^{loc}(\mathbb{R}^p) \times \mathbb{R}^{n_c} \to \mathcal{L}_1^{loc}(\mathbb{R}^m)$ that maps the measurement signal $\boldsymbol{y} \in \mathcal{L}_1^{loc}(\mathbb{R}^q)$, the reference input $\boldsymbol{r} \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$, and an initial condition $\boldsymbol{x}_{c0} \in \mathbb{R}^n$ to a control input signal

$$\boldsymbol{u}_c = \Omega_C(\boldsymbol{y}_c, \boldsymbol{r}, \boldsymbol{x}_{c0}),\ \boldsymbol{u}_c \in \mathcal{L}_1^{loc}(\mathbb{R}^m). \tag{3.6}$$

The nominal controller (3.5) attached to the nominal plant (3.2) gives rise to the nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$, with that is associated the operator $\Omega_L : \mathcal{L}_1^{loc}(\mathbb{R}^p) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \times \mathbb{R}^{n_c} \to \mathcal{L}_1^{loc}(\mathbb{R}^p)$ that maps the reference signal $\boldsymbol{r} \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$, the disturbance input $\boldsymbol{d} \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and initial conditions $\boldsymbol{x}_0 \in \mathbb{R}^n$ and $\boldsymbol{x}_{c0} \in \mathbb{R}^{n_c}$ to the output

$$z = \Omega_L(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{x}_{c0}),\ z \in \mathcal{L}_1^{loc}(\mathbb{R}^p). \tag{3.7}$$

**Assumption 3.1.** *The nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$ formed by the nominal nonlinear plant (3.2) and the nominal nonlinear controller (3.5) is ISS with respect to the reference input $\boldsymbol{r}$ and the disturbance input $\boldsymbol{d}$. Furthermore, the tracking error*

$$\boldsymbol{e}_z(t) \triangleq \boldsymbol{r}(t) - z(t) \tag{3.8}$$

*satisfies desirable steady-state tracking, performance, and disturbance rejection properties for arbitrary initial conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_{c0}$.*

The desirable closed-loop properties are not further specified, since they are application-dependent. The performance specifications may involve overshoot limits, settling times, or $\mathcal{L}_2$ gains for the transient response, and the tracking properties may involve exact steady-state tracking or impose nonzero limits on the steady-state tracking error, each with or without disturbances. Whatever the nominally attained tracking and performance objectives are, the goal of reconfigurable control consists in recovering these nominal properties.

## 3.4 Faults in Nonlinear Systems

According to the IFAC Technical Committee SAFEPROCESS and DIN 40041 [46], a fault is ”an unpermitted deviation of at least one characteristic property of the system from the acceptable, usual, standard condition”. In other words, a fault causes the system behavior to deviate from its desired behavior in such a way that the system cannot fully serve its purpose with the fault. A failure is "a permanent interruption of a system’s ability to perform a required function under specified operating

conditions". Therefore, every failure is also a fault, but not vice versa. In this monograph, both faults and failures are considered and modelled in closely related ways. It is assumed that the faults and the failures appear abruptly and remain effective once they have occurred. From here on, when referring to faults, it is tacitly understood that faults and failures are meant.

Faults and failures may occur on multiple levels, namely on the component level, at the aggregate system level that includes all system components, and at any intermediate aggregation level that includes a subset of the system components. In this monograph, faults and failures at the level of actuator and sensor components are considered, and the goal of fault-tolerant control consists in preventing component-level faults and failures to develop into system-level failures.

In the literature on fault-tolerant control, two alternative fault models prevail. In additive fault models, the fault is considered as a disturbance-like input that is added to the state-space system equation [83]. However, the class of severe faults that may cause a loss of closed-loop stability is better represented by multiplicative models, since additive faults cannot lead to a loss of closed-loop stability if the closed-loop system is ISS [141]. Therefore, the multiplicative fault modelling framework is adopted in this monograph. It is remarked that additive fault models are well-suited for fault diagnosis purposes [141].

**Definition 3.3 (Actuator faults in nonlinear systems).** An *actuator fault* is an event occurring at time $t_f$ that changes the function $\boldsymbol{f}(\cdot,\cdot,\cdot)$ to the faulty function $\boldsymbol{f}_f(\cdot,\cdot,\cdot)$ of the same dimensions, where $\boldsymbol{f}_f(\cdot,\boldsymbol{0},\cdot) = \boldsymbol{f}(\cdot,\boldsymbol{0},\cdot)$. ◇

In other words, an actuator fault cannot change the autonomous behaviour of the nonlinear system, nor can it change the disturbance effect.

**Definition 3.4 (Sensor faults in nonlinear systems).** A *sensor fault* is an event occurring at time $t_f$ that changes the nominal measurement function $\boldsymbol{h}(\cdot)$ to the faulty measurement function $\boldsymbol{h}_f(\cdot)$ of the same dimensions. ◇

Typical technological examples for faults are stuck valves, failed motors, failed sensors, and changed characteristics of such devices. They lead to the faulty nonlinear system

$$\Sigma_{Pf}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{f}_f(\boldsymbol{x}_f(t), \boldsymbol{u}_f(t), \boldsymbol{d}(t)) \\ \boldsymbol{y}_f(t) &= \boldsymbol{h}_f(\boldsymbol{x}_f(t)) \\ \boldsymbol{z}_f(t) &= \boldsymbol{h}_z(\boldsymbol{x}_f(t)) \end{cases} \qquad \boldsymbol{x}_f(0) = \boldsymbol{x}_0. \tag{3.9}$$

Associated with any faulty dynamical system (nonlinear, linear, Hammerstein-Wiener, piecewise affine) is a dynamical operator $\Omega_{Pf}: \mathcal{L}_1^{loc}(\mathbb{R}^m) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \to \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathcal{L}_1^{loc}(\mathbb{R}^p)$ that maps input signals $\boldsymbol{u}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^m)$, $\boldsymbol{d} \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and an initial condition $\boldsymbol{x}_0 \in \mathbb{R}^n$ to output signals $\boldsymbol{y}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^q)$, $\boldsymbol{z}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$,

$$(\boldsymbol{y}_f, \boldsymbol{z}_f) = \Omega_{Pf}(\boldsymbol{u}_f, \boldsymbol{d}, \boldsymbol{x}_0). \tag{3.10}$$

The operator will be useful in the general definition of reconfigurable control problems in the following section.

The occurrence of faults at the time $t_f$ changes the nominal plant to the faulty plant whose model was stated in Section 3.4 above. The reconfigured controller is determined as soon as the diagnostic result is available, but its synthesis requires some time. Without loss of generality, the time $t_r$ when the reconfigured controller becomes available is zero, $t_r = 0$. Therefore, the event times of the fault, availability of the diagnosis results, and the availability of the reconfigured controller are ordered in the sequence $t_f < t_D < t_r = 0$. These relations are shown in Fig. 3.3, where the shown sample trajectory is governed by nominal closed-loop dynamics $\Sigma_L = (\Sigma_P, \Sigma_C)$ for $t < t_f$. The deviation of the system state from the specifications is unavoidable during the time interval $(t_f, 0)$.

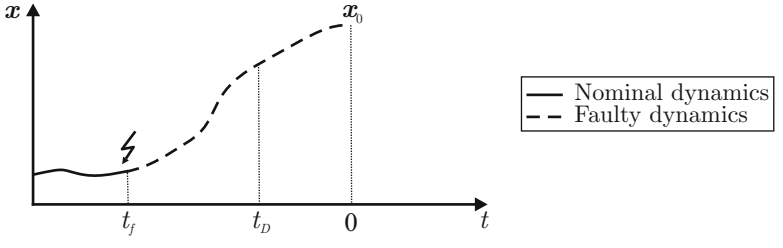


**Fig. 3.3** The faulty system deviates from the specifications. The solid solution is governed by nominal dynamics, whereas the dashed solution is governed by faulty dynamics.

**Example 3.2 (Nonlinear models of a ship subject to faults).** *In the nonlinear ship model (1.1)–(1.4), faults are introduced as follows. The failure of the yaw rate gyro sensor, fault $f_1$, leaves the system dynamics (1.1)–(1.4) unchanged but changes the output equations to*

$$y_{f1}(t) = v(t) \tag{3.11}$$

$$y_{f2}(t) = 0 \tag{3.12}$$

$$y_{f3}(t) = \psi(t), \tag{3.13}$$

*meaning that the gyro sensor outputs the value zero after its failure.*

*A blockage of the rudder $u_3$ in some blockage position $p \in [-1, 1]$ between maximum clockwise rudder moment and maximum counter-clockwise rudder moment, corresponding to the fault $f_2$, is modelled by the dynamical equations changed from (1.1)–(1.4) to*

$$\dot{v}_f(t) = \frac{m_{22}}{m_{11}} w_f(t) r_f(t) - \frac{d_{11}}{m_{11}} v_f(t) + \frac{1}{m_{11}} (u_{f1}(t) + u_{f2}(t)) + a_v(t) \tag{3.14}$$

$$\dot{w}_f(t) = -\frac{m_{11}}{m_{22}} v_f(t) r_f(t) - \frac{d_{22}}{m_{22}} w_f(t) + a_w(t) \tag{3.15}$$

$$\dot{r}_f(t) = \frac{m_{11} - m_{22}}{m_{33}} v_f(t) w_f(t) - \frac{d_{33}}{m_{33}} r_f(t) + \frac{1}{m_{33}} \left( \frac{b}{2} (u_{f1}(t) - u_{f2}(t)) + p \right) \tag{3.16}$$

$$\dot{\psi}_f(t) = r_f(t). \tag{3.17}$$

*The fault $f_3$ meaning a floating rudder is obtained by the special case $p = 0$.*

*It is physically clear that the compensation of a blocked rudder that causes a nonzero yaw moment requires asymmetric thrust forces, to which physical limits are set by the actuation range limits.*

*To illustrate the effect of faults on the ship behavior, simulations of the nonlinear model of the ship subject to a floating rudder ($f_3$) is shown in Figs. 3.4 and 3.5, where a proportional and integral control law is applied that succeeds in steering the fault-free ship clear of a physical obstacle. The controller uses the rudder to act on the yaw rate, and it uses the thrusters to act on the surge velocity. The loss of*

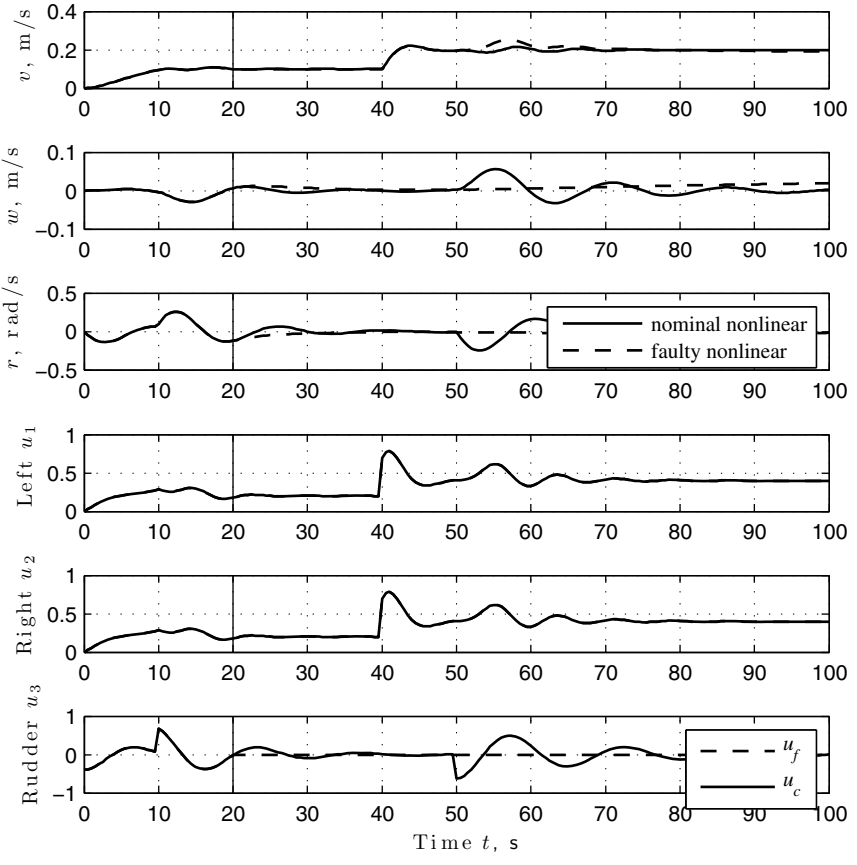


**Fig. 3.4** Nonlinear nominal and faulty ship responses in terms of the surge, sway, and yaw velocities ($f_3$).

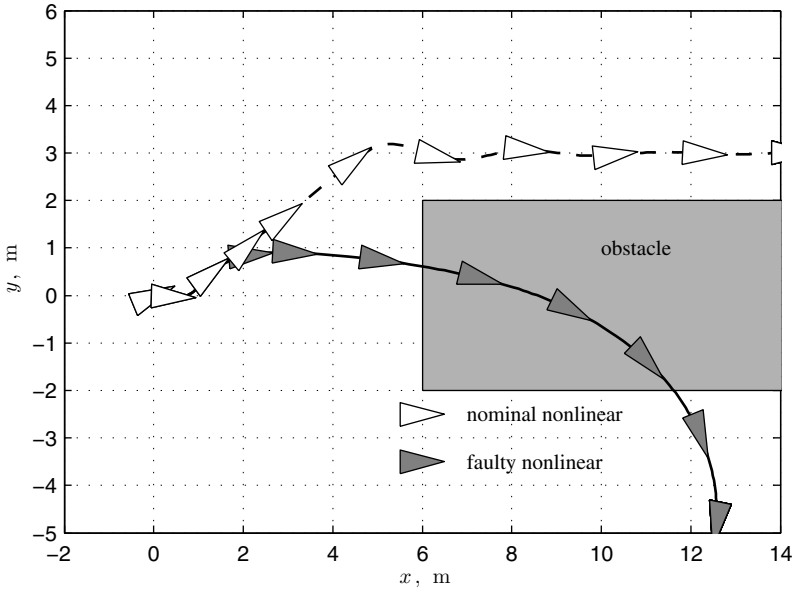


**Fig. 3.5** Nonlinear nominal and faulty ship responses in absolute coordinates ($f_3$).

*rudder effectiveness occurs at $t_f = 20$ s and is clearly visible in Fig. 3.4. Therefore, the yaw moment cannot be applied, and the ship course cannot be controlled after $t_f$, with the consequence that the ship runs into the obstacle (Fig. 3.5).*

While the nonlinear system could in principle represent every fault aspect of interest, the development of reconfigurable control approaches for general nonlinear systems is very hard. The difficulty is due to two major reasons. First, the reconfiguration methods must guarantee closed-loop properties like stability and tracking. Second, the reconfiguration methods must be capable of autonomous implementation on computing hardware. In other words, the reconfiguration involves automatic controller synthesis. The methods developed in this monograph for Hammerstein-Wiener systems and piecewise affine systems satisfy both requirements.

## 3.5 Reconfiguration Problems Based on the Model-Matching Idea

The standard approach to reconfigurable control consists in replacing the nominal controller $\Sigma_C$ with a new reconfigured controller

$$\Sigma_{Cr}: \begin{cases} \dot{\boldsymbol{x}}_{cr}(t) &= \boldsymbol{f}_{cr}(\boldsymbol{x}_{cr}(t), \boldsymbol{y}_f(t), \boldsymbol{r}(t)) \\ \boldsymbol{u}_f(t) &= \boldsymbol{h}_{cr}(\boldsymbol{x}_{cr}(t), \boldsymbol{y}_f(t), \boldsymbol{r}(t)), \\ \boldsymbol{x}_{cr}(0) &= \boldsymbol{x}_{cr0} \end{cases} \tag{3.18}$$

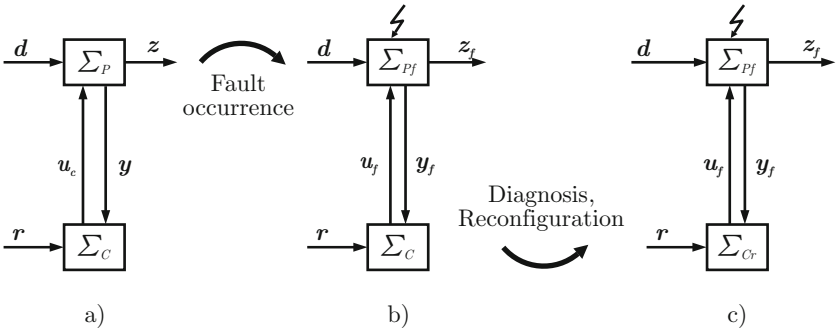


**Fig. 3.6** The model-matching approach to reconfigurable control: a) nominal closed-loop system, b) faulty plant controlled by nominal controller prior to reconfiguration, c) reconfigured closed-loop system with new controller.

at reconfiguration time, with that is associated an operator $\Omega_{Cr} : \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathcal{L}_1^{loc}(\mathbb{R}^p) \times \mathbb{R}^{n_c} \to \mathcal{L}_1^{loc}(\mathbb{R}^m)$ that maps the measurement signal $\boldsymbol{y}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^q)$, the reference input $\boldsymbol{r} \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$, and an initial condition $\boldsymbol{x}_{cr0} \in \mathbb{R}^{n_c}$ to a control input signal

$$\boldsymbol{u}_f = \Omega_{Cr}(\boldsymbol{y}_f, \boldsymbol{r}, \boldsymbol{x}_{cr0}),\ \boldsymbol{u}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^m). \tag{3.19}$$

Fig. 3.6 illustrates how the nominal plant $\Sigma_P$ in Fig. 3.6a) changes to the faulty plant $\Sigma_{Pf}$ in Fig. 3.6b) when the fault occurs. The reconfiguration step after fault diagnosis consists in the online synthesis and implementation of a reconfigured controller $\Sigma_{Cr}$ as shown in Fig. 3.6c).

The new controller gives rise to the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_{Cr})$ with that is associated the dynamical operator $\Omega_{Lr} : \mathcal{L}_1^{loc}(\mathbb{R}^p) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \times \mathbb{R}^{n_c} \to \mathcal{L}_1^{loc}(\mathbb{R}^p)$ that maps the reference signal $\boldsymbol{r} \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$, the disturbance input $\boldsymbol{d} \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and initial conditions $\boldsymbol{x}_0 \in \mathbb{R}^n$ and $\boldsymbol{x}_{cr0} \in \mathbb{R}^{n_c}$ to the output

$$\boldsymbol{z}_f = \Omega_{Lr}(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{x}_{cr0}),\ \boldsymbol{z}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^p). \tag{3.20}$$

Formally, the following reconfiguration goals are formulated, referring to the nonlinear system (3.2) and the dynamical operator (3.1) that is here used to represent the input/output mapping of the nonlinear plant.

**Problem 3.1 (Stability recovery).** Consider the nominal system $\Sigma_P$ defined in Equation (3.2), and the faulty system $\Sigma_{Pf}$ defined in Equation (3.9), both with the initial condition $\boldsymbol{x}_0$. Find a reconfigured controller $\Sigma_{Cr}$ of the form (3.18) such that

$$(\Sigma_{Pf}, \Sigma_{Cr}) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})$$

for any initial condition $\boldsymbol{x}_{cr0}$.

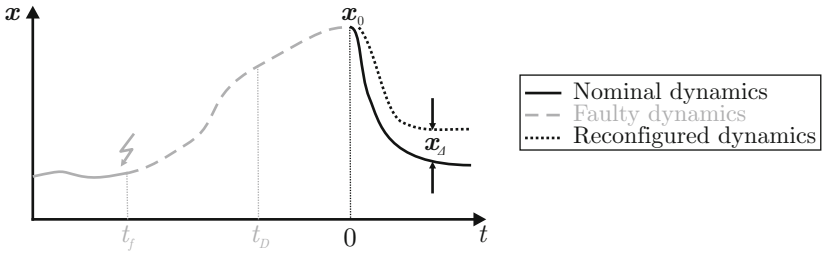


**Fig. 3.7** The reconfiguration problem starts at the initial time $t = 0$ and the initial condition $\boldsymbol{x}_0$.

This problem requires that the reconfigured closed-loop system be input-to-state stable with respect to its input $(\boldsymbol{r}, \boldsymbol{d})$.

**Problem 3.2 (Exact closed-loop performance and tracking recovery).** Consider the nominal system $\Sigma_P$ of the form (3.2) described by the operator (3.1), the faulty system $\Sigma_{Pf}$ of the form (3.9) described by the operator (3.10), both with the initial condition $\boldsymbol{x}_0$, and the nominal controller $\Sigma_C$ of the form (3.5) described by the operator (3.6) with initial condition $\boldsymbol{x}_{c0}$. Find a reconfigured controller $\Sigma_{Cr}$ described by the operator (3.19) with a suitable initial condition $\boldsymbol{x}_{cr0}$ such that

$$\Omega_{Lr}(\cdot, \cdot, \boldsymbol{x}_0, \boldsymbol{x}_{c0}) = \Omega_L(\cdot, \cdot, \boldsymbol{x}_0, \boldsymbol{x}_{cr0}).$$

This problem requires that the behaviour of the reconfigured closed-loop system exactly matches the nominal closed-loop behaviour. In other words, the nominal tracking properties, the nominal performance, and the nominal disturbance rejection properties are ideally recovered by the reconfigured closed-loop system.

Fig. 3.7 illustrates that the reconfigured controller starts with the faulty plant being in the state $\boldsymbol{x}(t_r) = \boldsymbol{x}(0) = \boldsymbol{x}_0$, since the system state deviates from the specifications. In the ideal case of perfect reconfiguration, the reconfigured system trajectories would follow nominal dynamics (solid in Fig. 3.7). Realistically, the reconfiguration is not ideal so that the reconfigured system trajectories differ from the nominal ones (dotted in Fig. 3.7). The difference signal is called $\boldsymbol{x}_\Delta$, see Fig. 3.7.

This general formulation of Problems 3.1 and 3.2 has been the basis of numerous prior approaches to reconfigurable control, see Chapter 1.6. The problem formulation is further refined in this monograph as described in the next section. The approach taken in this monograph employs an explicit model for the difference $\boldsymbol{x}_\Delta$.

## 3.6 Fault-Hiding Idea

Instead of directly solving the reconfiguration problems 3.1 and 3.2 through model matching, a special structure is imposed on the reconfigured controller $\Sigma_{Cr}$, which is factorised into the original nominal controller $\Sigma_C$ and a reconfiguration block $\Sigma_R$: $\Sigma_{Cr} = (\Sigma_R, \Sigma_C)$. The reconfiguration approach adopted here, therefore, consists in augmenting the closed loop by means of a dynamical reconfiguration block, as shown in Fig. 3.8.

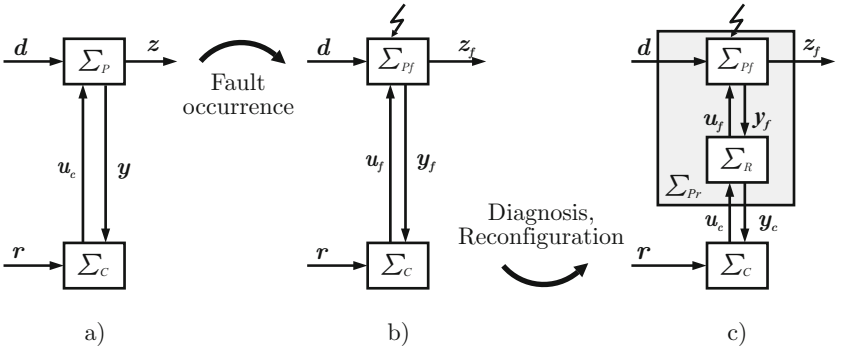


**Fig. 3.8** Fault-hiding concept: a) Nominal closed-loop system, b) faulty plant controlled by nominal controller prior to reconfiguration, c) reconfigured closed-loop system with reconfiguration block.

**Definition 3.5 (Nonlinear reconfiguration block).** A nonlinear dynamical system of the form

$$\Sigma_R : \begin{cases} \dot{\boldsymbol{\zeta}}(t) &= \boldsymbol{f}_{\boldsymbol{\zeta}}(\boldsymbol{\zeta}(t), \boldsymbol{u}_c(t), \boldsymbol{y}_f(t)) \\ \boldsymbol{u}_f(t) &= \boldsymbol{h}_{\boldsymbol{\zeta u}}(\boldsymbol{\zeta}(t), \boldsymbol{u}_c(t)) \\ \boldsymbol{y}_c(t) &= \boldsymbol{h}_{\boldsymbol{\zeta y}}(\boldsymbol{\zeta}(t), \boldsymbol{y}_f(t)) \end{cases} \tag{3.21}$$
$$\boldsymbol{\zeta}(0) = \boldsymbol{\zeta}_0,$$

where $\boldsymbol{\zeta}(t) \in \mathbb{R}^w$, is called a *nonlinear reconfiguration block* for a given nominal plant (3.2) and a corresponding faulty plant (3.9), if it connects the faulty plant (3.9) to the nominal controller (3.5) through the common signal pairs $(\boldsymbol{u}_f, \boldsymbol{y}_f)$ and $(\boldsymbol{u}_c, \boldsymbol{y}_c)$, and if it satisfies the *inactivity conditions*

$$\boldsymbol{f}_f(\cdot,\cdot,\cdot) = \boldsymbol{f}(\cdot,\cdot,\cdot) \Rightarrow \boldsymbol{u}_f(t) = \boldsymbol{u}_c(t) \tag{3.22}$$
$$\boldsymbol{h}_f(\cdot) = \boldsymbol{h}(\cdot) \Rightarrow \boldsymbol{y}_c(t) = \boldsymbol{y}_f(t) \tag{3.23}$$

for a suitable choice of its parameter functions $\boldsymbol{f}_{\boldsymbol{\zeta}}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}_{\boldsymbol{\zeta u}}(\cdot,\cdot)$, and $\boldsymbol{h}_{\boldsymbol{\zeta y}}(\cdot,\cdot)$. ⋄

The reconfiguration block translates the control input $\boldsymbol{u}_c$ from the nominal controller into a meaningful control input $\boldsymbol{u}_f$ for the faulty plant and corrects the measurement $\boldsymbol{y}_f$ from the faulty plant, so that the nominal controller sees the nominal plant behavior at the output $\boldsymbol{y}_c$ (Fig. 3.8c)). The inactivity conditions guarantee that the reconfiguration block does not affect the controller prior to faults. They ensure that the reconfiguration block may be implemented and run also during nominal plant operation, and its activation amounts to a change of its parameter functions $\boldsymbol{f}_{\boldsymbol{\zeta}}(\cdot,\cdot,\cdot)$, $\boldsymbol{h}_{\boldsymbol{\zeta u}}(\cdot,\cdot)$, and $\boldsymbol{h}_{\boldsymbol{\zeta y}}(\cdot,\cdot)$.

Associated with the reconfiguration block (3.21) is the operator $\Omega_R : \mathcal{L}_1^{loc}(\mathbb{R}^m) \times \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathbb{R}^w \to \mathcal{L}_1^{loc}(\mathbb{R}^m) \times \mathcal{L}_1^{loc}(\mathbb{R}^q)$,

$$(\boldsymbol{u}_f, \boldsymbol{y}_c) = \Omega_R(\boldsymbol{u}_c, \boldsymbol{y}_f, \boldsymbol{\zeta}_0). \tag{3.24}$$

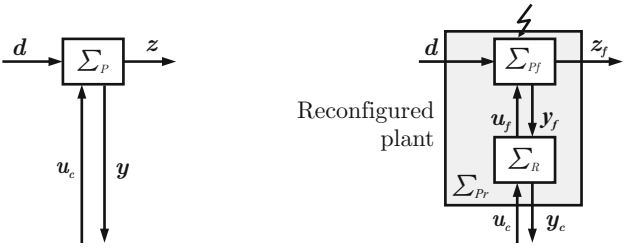


**Fig. 3.9** Due to the fault-hiding concept, the nominal controller is irrelevant for reconfiguration block design. The nominal plant a) and the reconfigured plant b) are compared in open-loop configuration.

Together with the faulty plant (9.7), the reconfiguration block (3.24) forms the *reconfigured plant* $\Sigma_{Pr} = (\Sigma_{Pf}, \Sigma_R)$ described by the operator $\Omega_{Pr} : \mathcal{L}_1^{loc}(\mathbb{R}^m) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \times \mathbb{R}^w \to \mathcal{L}_1^{loc}(\mathbb{R}^q) \times \mathcal{L}_1^{loc}(\mathbb{R}^p)$

$$(\boldsymbol{y}_c, \boldsymbol{z}_f) = \Omega_{Pr}(\boldsymbol{u}_c, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{\zeta}_0), \tag{3.25}$$

which is connected to the nominal controller $\Sigma_C$ defined in Equation (3.5) by means of the signal pair $(\boldsymbol{u}_c, \boldsymbol{y}_c)$ (see Fig. 3.9). On the one hand, this point of view is useful for reasoning about appropriate synthesis of the reconfiguration block. Namely, the comparison of the nominal plant behaviour and the reconfigured plant behaviour will guide the synthesis of the reconfiguration block. On the other hand, from an implementation point of view, the combination $(\Sigma_R, \Sigma_C) = \Sigma_{Cr}$ represents the reconfigured controller $\Sigma_{Cr}$ that is implemented. Associated with the complete reconfigured closed-loop system $\Sigma_{LFH} = (\Sigma_{Pf}, \Sigma_R, \Sigma_C)$ is the dynamical operator $\Omega_{LFH} : \mathcal{L}_1^{loc}(\mathbb{R}^p) \times \mathcal{L}_1^{loc}(\mathbb{R}^k) \times \mathbb{R}^n \times \mathbb{R}^w \times \mathbb{R}^{n_c} \to \mathcal{L}_1^{loc}(\mathbb{R}^p)$ that maps the reference signal $\boldsymbol{r} \in \mathcal{L}_1^{loc}(\mathbb{R}^p)$, the disturbance input $\boldsymbol{d} \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and initial conditions $\boldsymbol{x}_0 \in \mathbb{R}^n$, $\boldsymbol{x}_{cr0} \in \mathbb{R}^{n_c}$, and $\boldsymbol{\zeta}_0 \in \mathbb{R}^w$ to the output

$$\boldsymbol{z}_f = \Omega_{LFH}(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{\zeta}_0, \boldsymbol{x}_{cr0}),\ \boldsymbol{z}_f \in \mathcal{L}_1^{loc}(\mathbb{R}^p). \tag{3.26}$$

The following goals ensure that the original controller "sees" the fault-free plant behavior when attached to the reconfigured plant. They are deliberately introduced because they are key for keeping the nominal controller a part of the overall reconfigured closed-loop system. They will be instrumental in solving Problems 3.1 and 3.2.

**Definition 3.6 (Weak fault-hiding goal).** The reconfigured plant $\Sigma_{Pr}$ satisfies the weak fault-hiding goal, if the relation

$$\forall \boldsymbol{x}_0,\ \exists \boldsymbol{\zeta}_0 : \Omega_{Pr}^{\boldsymbol{y}_c}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0, \boldsymbol{\zeta}_0) - \Omega_P^{\boldsymbol{y}}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0) = \boldsymbol{0}$$

holds. ◇

In words, for every plant initial condition $\boldsymbol{x}_0$, there must exist a matching initial condition $\boldsymbol{\zeta}_0$ of the reconfiguration block, such that the reconfigured plant behaviour is identical to the fault-free plant behaviour in the absence of disturbances. The term "weak" fault-hiding indicates that the initial state $\boldsymbol{\zeta}_0$ of the reconfiguration block depends on the initial state $\boldsymbol{x}_0$ of the faulty plant. Furthermore, the disturbance behavior is not required to be equal in the nominal and reconfigured plants. The weak fault-hiding goal is used in Parts II and III of this monograph.

The weak fault-hiding goal is a relaxation of the following asymptotic fault-hiding goal.

**Definition 3.7 (Asymptotic fault-hiding goal).** The reconfigured plant $\Sigma_{Pr}$ satisfies the asymptotic fault-hiding goal, if the relations

$$\begin{aligned} \forall \boldsymbol{x}_0,\ \exists \hat{\boldsymbol{\zeta}}_0 : \Omega^{\boldsymbol{y}_c}_{Pr}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0, \hat{\boldsymbol{\zeta}}_0) - \Omega^{\boldsymbol{y}}_{P}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0) = \boldsymbol{0} \\ \forall \boldsymbol{x}_0,\ \forall \tilde{\boldsymbol{\zeta}}_0 \neq \hat{\boldsymbol{\zeta}}_0 :\ \lim_{t\to\infty} \left( \Omega^{\boldsymbol{y}_c}_{Pr}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0, \tilde{\boldsymbol{\zeta}}_0) - \Omega^{\boldsymbol{y}}_{P}(\cdot, \boldsymbol{0}, \boldsymbol{x}_0) \right) = \boldsymbol{0} \end{aligned} \tag{3.27}$$

hold. ◇

In other words, in addition to weak fault-hiding, asymptotic fault-hiding requires that for all nonmatching initialisations of the reconfiguration block, the difference between nominal and reconfigured plant measurements vanishes asymptotically. This goal is used in Part II of this monograph.

A stronger version of the asymptotic fault-hiding goal is given by the strict fault-hiding goal used in Chapter 4 and Part II, where $\boldsymbol{\zeta}_0$ is not allowed to depend on $\boldsymbol{x}_0$, and where, in addition, also the disturbance behaviour must be recovered.

**Definition 3.8 (Strict fault-hiding goal).** The reconfigured plant $\Sigma_{Pr}$ satisfies the strict fault-hiding goal, if the relation

$$\exists \hat{\boldsymbol{\zeta}}_0 \text{ such that } \forall \boldsymbol{x}_0 : \Omega^{\boldsymbol{y}_c}_{Pr}(\cdot, \cdot, \boldsymbol{x}_0, \hat{\boldsymbol{\zeta}}_0) - \Omega^{\boldsymbol{y}}_{P}(\cdot, \cdot, \boldsymbol{x}_0) = \boldsymbol{0} \tag{3.28}$$

holds. ◇

It follows from the definitions that strict fault-hiding implies asymptotic fault-hiding, which in turn implies weak fault-hiding. In the fault-hiding approach a reconfiguration block is sought that satisfies one of the fault-hiding goals. The approach offers the following advantages over the redesign approach:

- If the nominal controller is a human operator, e.g. a pilot, then the fault-hiding approach reduces the difficulty linked to dealing with a faulty system, because the reconfigured system behaves like the nominal system. As a consequence, it reduces training efforts for large numbers of fault scenarios and stress during fault situations.
- The design of the reconfiguration block $\Sigma_R$ is independent of the controller and, therefore, usable with any nominal controller (consider different people taking shifts in operating the plant).

- The fault-hiding strategy opens the way for minimum-invasive alterations of the loop. If the controller is automatic and the fault affects small parts of the plant only, then large parts of the nominal controller are still valid and should be kept instead of performing a complete redesign, which may be costly and time-consuming.

The reconfiguration block is determined during its synthesis such that the fault is hidden from the nominal controller. Furthermore, additional closed-loop objectives strongly influence its synthesis, which are described next.

## 3.7 Stability, Tracking, and Performance Recovery in the Fault-Hiding Approach

The definitions of the fault-hiding goals have been formulated in open loop. Nevertheless, the final objective in reconfigurable control consists in the recovery of desirable properties of the nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$ by the reconfigured closed-loop system $\Sigma_{LFH} = (\Sigma_{Pf}, \Sigma_R, \Sigma_C)$. Therefore, the Problems 3.1 and 3.2 are next adapted to the fault-hiding context; the exact performance recovery problem is also split into the aspects of setpoint tracking and performance recovery. The first important property is stability.

**Problem 3.3 (Stability recovery by fault-hiding).** Consider the nominal plant $\Sigma_P$ defined in Equation (3.2) and the faulty system $\Sigma_{Pf}$ defined in Equation (3.9). Find a reconfiguration block $\Sigma_R$ of the form (3.21) such that for all nominal controllers $\Sigma_C$ that satisfy Assumption 3.1,

$$\{(\Sigma_P, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\}.$$

This problem requires that the reconfigured closed-loop system be input-to-state stable whenever the nominal closed-loop system has the same property. Furthermore, performance and tracking are important properties to be recovered. Setpoint tracking refers to the ability of the system to asymptotically follow a constant reference signal in steady-state.

**Problem 3.4 (Asymptotic setpoint tracking recovery by fault-hiding).** Consider the nominal plant $\Sigma_P$ with the initial condition $\boldsymbol{x}_0$ described by the operator (3.1) and the faulty system $\Sigma_{Pf}$ with the initial condition $\boldsymbol{x}_0$ associated with the operator (3.10). Let Assumption 3.1 be satisfied, and let

$$z_f = \Omega_{LFH}(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{\zeta}_0, \boldsymbol{x}_{c0}) \tag{3.29}$$

$$z = \Omega_L(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{x}_0, \boldsymbol{x}_{c0}) \tag{3.30}$$

be the outputs of the reconfigured and the nominal closed-loop systems. Find a reconfiguration block $\Sigma_R$ described by the operator (3.24) such that for all admissible controllers $\Sigma_C$, it follows that

$$\forall \boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t), \boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \lim_{t\to\infty} \Big(\boldsymbol{z}_f(t) - \boldsymbol{z}(t)\Big) = \boldsymbol{0}. \tag{3.31}$$

Performance refers to overshoot and settling time properties of transients.

**Problem 3.5 (Exact performance recovery by fault-hiding).** Consider the nominal plant $\Sigma_P$ with the initial condition $\boldsymbol{x}_0$ described by the operator (3.1), the faulty system $\Sigma_{Pf}$ with the initial condition $\boldsymbol{x}_0$ described by the operator (3.10). Let $\Sigma_C$ be an admissible nominal controller $\Sigma_C$ with the initial condition $\boldsymbol{x}_{c0}$ described by the operator (3.6), in the sense that $\Sigma_C$ satisfies Assumption 3.1. Find a reconfiguration block $\Sigma_R$ with the initial condition $\boldsymbol{\zeta}_0$ described by the operator (3.24) such that for all admissible controllers, it follows that

$$\Omega_{LFH}(\cdot,\cdot,\cdot,\boldsymbol{\zeta}_0,\cdot) - \Omega_L(\cdot,\cdot,\cdot,\cdot) = \boldsymbol{0}. \tag{3.32}$$

Note that all three previous problems are formulated independent of any specific controller. By this formulation, reconfiguration solutions are sought that work with *any* admissible controller satisfying Assumption 3.1.

## 3.8 Basic Structure of Fault-Hiding Solutions

This section presents a general structure for the reconfiguration block (3.21) for reconfiguration after actuator as well as sensor faults that is used throughout this monograph. Its particular realisation varies with the system class and the considered types of faults. Minor extensions of this structure will be made, but the underlying idea is well illustrated in the following simplified way.

The idea shown in Fig. 3.10 consists in realising the reconfiguration block $\Sigma_R$ by means of a virtual sensor $\Sigma_S$ and a virtual actuator $\Sigma_A$. The virtual sensor $\Sigma_S$ consists of a model for the faulty system augmented by output injection, and the measurement from the faulty plant is replaced by its estimate. The virtual actuator $\Sigma_A$ consists of a reference model for the nominal plant as well as feedback of the difference between the state of the reference model and the state estimate of the observer and feedforward of the control input $\boldsymbol{u}_c$. In summary, $\Sigma_R = (\Sigma_S, \Sigma_A)$.

**Definition 3.9 (Nonlinear virtual sensor).** The *nonlinear virtual sensor* is the nonlinear dynamical system

$$\Sigma_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= \boldsymbol{f}_f(\hat{\boldsymbol{x}}_f(t), \boldsymbol{u}_f(t), \boldsymbol{0}) + \boldsymbol{L}(\boldsymbol{y}_f(t) - \hat{\boldsymbol{y}}_f(t)) \\ \hat{\boldsymbol{y}}_f(t) &= \boldsymbol{h}_f(\hat{\boldsymbol{x}}_f(t)) \end{cases} \tag{3.33}$$

with the initial condition $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$. ◇

The virtual sensor contains a copy of the faulty plant model (3.9) for zero disturbance (since the disturbance is generally not available for measurement) augmented by linear output error injection through the matrix gain $\boldsymbol{L}$.

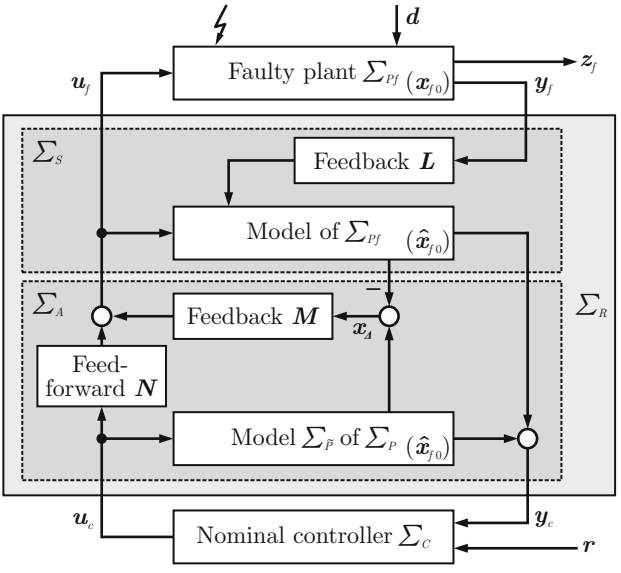


**Fig. 3.10** Structure of the activated reconfiguration block.

**Definition 3.10 (Nonlinear virtual actuator).** The *nonlinear virtual actuator* is the nonlinear dynamical system

$$\Sigma_A : \begin{cases} \dot{\tilde{x}}(t) & = f(\tilde{x}(t), u_c(t), 0) \\ u_f(t) & = M x_\Delta(t) + N u_c(t) \\ y_c(t) & = h(\tilde{x}(t)) \end{cases} \tag{3.34}$$

with the difference state

$$x_\Delta(t) \triangleq \tilde{x}(t) - \hat{x}_f(t) \tag{3.35}$$

and the initial condition $\tilde{x}(0) = \hat{x}_{f0}$. ◇

The virtual actuator contains a reference model

$$\Sigma_{\tilde{P}} : \begin{cases} \dot{\tilde{x}}(t) & = f(\tilde{x}(t), u_c(t), 0) \\ \tilde{x}(0) & = \hat{x}_{f0} \end{cases} \tag{3.36}$$

for the nominal plant (3.9) with the state $\tilde{x}$ for zero disturbance. Furthermore, linear state feedback of the difference state (3.35) through the matrix gain $M$ and feed-forward of the control input $u_c$ through the matrix gain $N$ to the plant input $u_f$ are

applied. The virtual actuator outputs the nominal measurement $\boldsymbol{y}_c$. In order to satisfy the inactivity conditions (3.22), (3.23) of the reconfiguration block definition, the virtual actuator equations are modified to produce $\boldsymbol{u}_f(t) = \boldsymbol{u}_c(t)$ and $\boldsymbol{y}_c(t) = \boldsymbol{y}_f(t)$ before the detection and isolation of any faults.

Both the observer and the virtual actuator involve linear feedback and feedforward gains $\boldsymbol{L}$, $\boldsymbol{M}$, and $\boldsymbol{N}$, which are free to be designed in a suitable way such that as many as possible of the Problems 3.3–3.5 are solved. The use of more general nonlinear feedback and feedforward laws is conceivable, but not considered in this monograph, bearing in mind the requirement that the reconfigured controller must be found autonomously and online as fast as possible (recall from Chapter 1.3, page 7 that due to $t_f < t_D < 0$, the faulty system generally deviates from its designated operating regime until the reconfigured controller is available at $t = 0$).

Both the observer (3.33) and the virtual actuator are provided with the same initial condition $\hat{\boldsymbol{x}}_{f0}$ at reconfiguration time $t = 0$. Ideally, the true plant state at that time is used. Since the state is generally not available for measurement, the initial condition must be guessed. The initial guess can be improved by running the observer along with the nominal system prior to the detection of any faults. However, all solutions to Problems 3.3–3.5 provided in this monograph are valid whether or not the guessed initial condition is accurate, because none of the fault-hiding goals requires the knowledge of the plant initial condition, but only that if it was known, a matching initial condition for the reconfiguration block can be found.

**Example 3.3 (Nonlinear virtual sensor and nonlinear virtual actuator for the ship).** *Consider the ship model (1.1)–(1.4). The failure $f_1$ of the gyro sensor modifies the model to the faulty ship output equations (3.11)–(3.13), and the blockage $f_2$ of the rudder $u_3$ at the position* 0.1 *modifies the model to (3.14)–(3.17), where $p = 0.1$. All true input and output variables for the faulty ship are denoted by subscript $f$. A nonlinear virtual sensor for the ship subject to the faults $f_1, f_2$ has the form*

$$\Sigma_S : \begin{cases} \dot{\hat{v}}(t) & = \frac{m_{22}}{m_{11}}\hat{w}(t)\hat{r}(t) - \frac{d_{11}}{m_{11}}\hat{v}(t) + \frac{1}{m_{11}}(u_{f,1}(t) + u_{f,2}(t)) + \boldsymbol{l}_1(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}_f(t)) \\ \dot{\hat{w}}(t) & = -\frac{m_{11}}{m_{22}}\hat{v}(t)\hat{r}(t) - \frac{d_{22}}{m_{22}}\hat{w}(t) + \boldsymbol{l}_2(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}_f(t)) \\ \dot{\hat{r}}(t) & = \frac{m_{11}-m_{22}}{m_{33}}\hat{v}(t)\hat{w}(t) - \frac{d_{33}}{m_{33}}\hat{r}(t) + \frac{1}{m_{33}}\left(\frac{b}{2}(u_{f,1}(t) - u_{f,2}(t) + 0.1)\right) + \boldsymbol{l}_3(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}_f(t)) \\ \dot{\hat{\psi}}(t) & = \hat{r}(t) + \boldsymbol{l}_3(\boldsymbol{y}(t) - \hat{\boldsymbol{y}}_f(t)) \\ \hat{\boldsymbol{y}}_f(t) & = \left(\hat{v}(t)\ 0\ 0\ \hat{\psi}(t)\right)^T \end{cases} \tag{3.37}$$

*where the rudder input $u_3$ has been replaced by the constant* 0.1*, and where the gain*

$$\boldsymbol{L} = \begin{pmatrix} \boldsymbol{l}_1 \\ \boldsymbol{l}_2 \\ \boldsymbol{l}_3 \\ \boldsymbol{l}_4 \end{pmatrix}$$

*must stabilise the observation error. The model parameter $u_{f,3} = 0.1$ and the output matrix must be provided by a fault diagnosis component. A nonlinear virtual actuator for the ship subject to the faults $f_1, f_2$ has the form*

$$\Sigma_A : \begin{cases} \dot{\tilde{v}}(t) & = \frac{m_{22}}{m_{11}}\tilde{w}(t)\tilde{r}(t) - \frac{d_{11}}{m_{11}}\tilde{v}(t) + \frac{1}{m_{11}}(u_{c,1}(t) + u_{c,2}(t)) \\ \dot{\tilde{w}}(t) & = -\frac{m_{11}}{m_{22}}\tilde{v}(t)\tilde{r}(t) - \frac{d_{22}}{m_{22}}\tilde{w}(t) \\ \dot{\tilde{r}}(t) & = \frac{m_{11}-m_{22}}{m_{33}}\tilde{v}(t)\tilde{w}(t) - \frac{d_{33}}{m_{33}}\tilde{r}(t) + \frac{1}{m_{33}}\left(\frac{b}{2}(u_{c,1}(t) - u_{c,2}(t) + u_{c,3}(t))\right) \\ \dot{\tilde{\psi}}(t) & = \tilde{r}(t) \\ \boldsymbol{u}_f(t) & = \boldsymbol{M}\begin{pmatrix} \tilde{\boldsymbol{v}}(t) - \hat{v}(t) \\ \tilde{w}(t) - \hat{w}(t) \\ \tilde{r}(t) - \hat{r}(t) \\ \tilde{\psi}(t) - \hat{\psi}(t) \end{pmatrix} + \boldsymbol{N}\begin{pmatrix} u_{c,1}(t) \\ u_{c,2}(t) \\ u_{c,3}(t) \end{pmatrix} \\ \boldsymbol{y}_c(t) & = \left(\tilde{v}(t)\ \tilde{w}(t)\ \tilde{r}(t)\ \tilde{\psi}(t)\right)^T, \end{cases} \tag{3.38}$$

*where the gains $\boldsymbol{M}$ and $\boldsymbol{N}$ must be chosen such that the difference system is stabilised. Furthermore, the tracking recovery is highly important especially for the heading variable $\psi$, since accurate steering is needed for safe collision and obstacle-avoidance maneuvers.*

*The control commands $u_{c,1}$, $u_{c,2}$, $u_{c,3}$ that are issued by a helmsman or an autopilot are translated into the control inputs $u_{f,1}$, $u_{f,2}$, $u_{f,3}$ that act on the ship. The fault-hiding concept has the advantage that the reconfiguration block $(\Sigma_S, \Sigma_A)$ is useful both for ships under manual control and for ships under autopilot control. Under manual control, the crew at the helm is relieved from the need to learn how to deal with the faulty ship, since the virtual actuator and virtual sensor take care of that task.*

## 3.9 General Properties of Fault-Hiding Solutions

Although special cases of the general nonlinear problem will be solved in Parts II and III, it is instructive to also study the following general properties of the adopted realisation of the reconfiguration block. It will turn out that the fault-hiding goals are always satisfied as a direct consequence of the chosen structure of the reconfiguration block decomposed into a virtual sensor and a virtual actuator. Furthermore, the following analysis reveals how the main reconfiguration problems regarding stability, tracking, and performance recovery objectives are reformulated. The new formulation points towards solution strategies that will be followed in this monograph.

The important properties are deduced by calculating the dynamics of the reconfigured plant $\Sigma_{Pr} = (\Sigma_{Pf}, \Sigma_S, \Sigma_A)$. These dynamics are originally formulated in terms of the state variables $\tilde{\boldsymbol{x}}$, $\hat{\boldsymbol{x}}_f$, and $\boldsymbol{x}_f$. Introducing the observation error

$$\boldsymbol{e}(t) \triangleq \hat{\boldsymbol{x}}_f(t) - \boldsymbol{x}_f(t), \tag{3.39}$$

the original state variables are expressed in new variables $\tilde{\boldsymbol{x}}$, $\boldsymbol{e}$, and $\boldsymbol{x}_\Delta$ that are linked to the original variables by the linear transformation

$$\begin{pmatrix} \tilde{\boldsymbol{x}} \\ \boldsymbol{e} \\ \boldsymbol{x}_\Delta \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 1 & -1 & 0 \end{pmatrix}\begin{pmatrix} \tilde{\boldsymbol{x}} \\ \hat{\boldsymbol{x}}_f \\ \boldsymbol{x}_f \end{pmatrix}.$$

The observation error dynamics is governed by the equation

$$\Sigma_e : \begin{cases} \dot{\boldsymbol{e}}(t) & = \boldsymbol{f}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t), \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t), \boldsymbol{0}) - \boldsymbol{L}\boldsymbol{h}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)) \\ & \quad - \Big[\boldsymbol{f}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t), \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t), \boldsymbol{d}(t)) \\ & \quad\quad - \boldsymbol{L}\boldsymbol{h}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t))\Big], \end{cases} \tag{3.40}$$

with the initial condition $\boldsymbol{e}(0) = \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0$, which is obtained from the definition (3.39) of the observation error and from the definition (3.33) of the nonlinear virtual sensor.

The dynamics of the difference system (3.35) is governed by the equation

$$\Sigma_{\Delta} : \begin{cases} \dot{\boldsymbol{x}}_{\Delta}(t) & = \boldsymbol{f}(\tilde{\boldsymbol{x}}(t), \boldsymbol{u}_c(t), \boldsymbol{0}) - \boldsymbol{f}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t), \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t), \boldsymbol{0}) \\ & \quad + \boldsymbol{L}\boldsymbol{h}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)) - \boldsymbol{L}\boldsymbol{h}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t)), \end{cases} \tag{3.41}$$

with the initial condition $\boldsymbol{x}_{\Delta}(0) = \boldsymbol{0}$, which is obtained from the definition (3.35) of the difference state and from the definition (3.34) of the nonlinear virtual actuator, and their initial conditions.

The combined dynamics (3.40), (3.41) together with the reference model in the virtual actuator (3.34) provide a transformed representation of the dynamics of the reconfigured plant (3.25). It is immediate from the virtual actuator definition (3.34) and from Fig. 3.10 that the plant behavior seen through the input $\boldsymbol{u}_c$ and the output $\boldsymbol{y}_c$ is nominal for zero disturbance. In other words, the state $\tilde{\boldsymbol{x}}$ is governed by the same dynamics as the nominal system state $\boldsymbol{x}(t)$. Therefore, at least the weak fault-hiding goal is satisfied by the adopted approach for any matrix gains $\boldsymbol{L}, \boldsymbol{M}, \boldsymbol{N}$. The fault-hiding property follows, therefore, by construction from the chosen reconfiguration block structure. Stronger fault-hiding properties depend on additional properties of the reconfigured plant, which must be achieved by means of appropriate synthesis of the matrix gains $\boldsymbol{L}, \boldsymbol{M}, \boldsymbol{N}$.

Furthermore, considering the reconfigured closed-loop system, the equations (3.40) and (3.41) together with the nominal controller (3.5) have a special structure that is illustrated in Fig. 3.11. Namely, the overall reconfigured closed-loop system has been transformed into the cascade interconnection of two feedback systems:

- the nominal closed-loop dynamics $(\Sigma_C, \Sigma_{\tilde{P}})$ (left dashed box in Fig. 3.11), ISS by assumption, and
- the feedback interconnection of the observation error dynamics defined in Equation (3.40) and denoted by $\Sigma_e$ and the difference system dynamics defined in Equation (3.41) and denoted by $\Sigma_{\Delta}$ (right dashed box in Fig. 3.11).

The two subsystems are connected through the coupling variables $\boldsymbol{u}_c$ and $\tilde{\boldsymbol{x}}$. The reference $\boldsymbol{r}$ and the disturbance $\boldsymbol{d}$ are external inputs.

Recall that the cascade connection of two ISS systems has the ISS property (Property 2.1). This observation has the following consequences, which well characterise the remaining tasks for the following chapters concerning the stability recovery (Problem 3.3).

1. The cascade connection provides a nonlinear separation property regarding the ISS of the reconfigured closed-loop dynamics. Namely, the ISS property of the

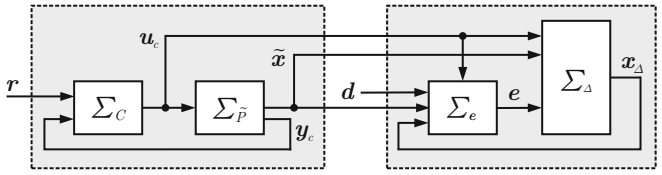


**Fig. 3.11** The transformed structure of the reconfigured closed-loop system.

reconfigured closed-loop system follows if the nominal closed-loop system is ISS (Assumption 3.1), and if the interconnection $(\Sigma_e, \Sigma_\Delta)$ of introduced dynamics is ISS.

2. The stability properties of the interconnection $(\Sigma_e, \Sigma_\Delta)$ are independent of the nominal controller $\Sigma_C$, as the inspection of Equations (3.40) and (3.41) shows.
3. The synthesis of the reconfiguration block amounts to finding the observer gain $\boldsymbol{L}$ and the virtual actuator gains $\boldsymbol{M}$ and $\boldsymbol{N}$. Due to aspect 2, the synthesis depends only on the models of the nominal and faulty plants.

In other words, solving Problem 3.3 reduces to stabilising the feedback interconnection $(\Sigma_e, \Sigma_\Delta)$. Finding stabilising gains $\boldsymbol{L}$, $\boldsymbol{M}$, and $\boldsymbol{N}$ and thus shaping the dynamics of $(\Sigma_e, \Sigma_\Delta)$ will be one of the central aspects of the chapters to follow. Assumptions about nominal closed-loop stability similar to Assumption 3.1 will be standing assumptions in later chapters.

Other important questions concern the recovery of the nominal tracking (Problem 3.4) and performance (Problem 3.5) properties by the reconfigured closed-loop system. The tracking error defined in Equation (3.8) for the nominal closed-loop system $\Sigma_L$ is obtained as follows

$$\boldsymbol{e}_z(t) = \boldsymbol{r}(t) - \boldsymbol{z}(t) = \boldsymbol{r}(t) - \boldsymbol{h}_z(\boldsymbol{x}(t)),$$

whereas the tracking error of the reconfigured closed-loop system $\Sigma_{LFH}$ is defined as

$$\begin{aligned}\boldsymbol{e}_{z_f}(t) &\triangleq \boldsymbol{r}(t) - \boldsymbol{z}_f(t) = \boldsymbol{r}(t) - \boldsymbol{h}_z(\boldsymbol{x}_f(t)) \\ &= \boldsymbol{r}(t) - \boldsymbol{h}_z(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t) - \boldsymbol{e}(t)).\end{aligned}$$

Note that the state $\tilde{\boldsymbol{x}}$ is governed by the same dynamics as the nominal system state $\boldsymbol{x}(t)$ except for the disturbance, which vanishes, and that Assumption 3.1 holds for arbitrary initial conditions. It follows that $\lim_{t\to\infty} \boldsymbol{r}(t) - \boldsymbol{h}_z(\tilde{\boldsymbol{x}}(t)) = \boldsymbol{0}$. Therefore, the asymptotic setpoint tracking properties are recovered if and only if

$$\lim_{t\to\infty} (\boldsymbol{x}_\Delta(t) + \boldsymbol{e}(t)) = \boldsymbol{0}$$

and thus

$$\lim_{t\to\infty} (\boldsymbol{e}_z(t) - \boldsymbol{e}_{z_f}(t)) = \boldsymbol{0}.$$

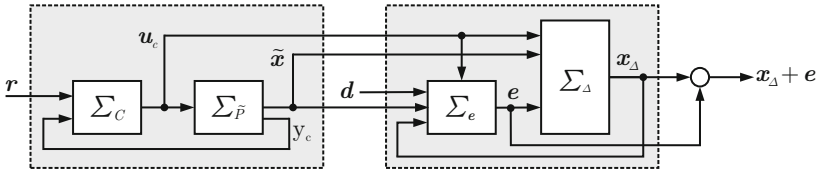


**Fig. 3.12** Reconfigurable control viewed as disturbance decoupling and output regulation problems.

As the case of exactly cancelling nonzero solutions is unlikely, the goal consists in independently regulating $\boldsymbol{x}_\Delta(t)$ and $\boldsymbol{e}(t)$ to the origin.

**Lemma 3.1.** *Problem 3.4 is solved, i.e. the nominal asymptotic closed-loop tracking properties are recovered by the reconfigured closed-loop system, if* $\forall \boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$ *(*$\bar{\boldsymbol{r}} \in \mathbb{R}^p$*),* $\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t)$ *(*$\bar{\boldsymbol{d}} \in \mathbb{R}^k$*) and* $\forall \boldsymbol{x}_0$, $\boldsymbol{x}_{c0}$, $\boldsymbol{\zeta}_0$

$$\lim_{t\to\infty} \boldsymbol{x}_\Delta(t) = \boldsymbol{0} \tag{3.42}$$

$$\lim_{t\to\infty} \boldsymbol{e}(t) = \boldsymbol{0}. \tag{3.43}$$

**Lemma 3.2.** *Problem 3.5 is solved, i.e. the nominal tracking and performance properties are exactly recovered, if* $\forall \boldsymbol{d}(t), \boldsymbol{r}(t)$ *and* $\forall \boldsymbol{x}_0$, $\boldsymbol{x}_{c0}$, $\boldsymbol{\zeta}_0$

$$\forall t \in \mathbb{R}_+ : \ \boldsymbol{x}_\Delta(t) + \boldsymbol{e}(t) = \boldsymbol{0}. \tag{3.44}$$

The basic structure (3.33), (3.34) of the reconfiguration block (3.21) is the basis for the reconfiguration solutions for linear systems, Hammerstein-Wiener systems, and piecewise affine systems. It will be modified and extended as necessary to accommodate the presence of disturbances and to recover the nominal tracking and performance properties in spite of disturbances by the reconfigured closed-loop system. However, the extensions always fit into the presented framework.

In other words,

1. the asymptotic setpoint tracking recovery Problem 3.4 reduces to ensuring that Equations (3.42) and (3.43) are satisfied, and
2. the exact performance recovery Problem 3.5 reduces to ensuring that Equation (3.44) is satisfied.

The tracking recovery problem described in Equations (3.42) and (3.43) can be interpreted as an output regulation problem with the output $\boldsymbol{x}_\Delta + \boldsymbol{e}$ (Fig. 3.12). The exact performance recovery described in Equation (3.44) can be interpreted as a disturbance decoupling problem with the input $(\boldsymbol{r}^T, \boldsymbol{d}^T)^T$ and the output $\boldsymbol{x}_\Delta + \boldsymbol{e}$. Solutions to the general nonlinear output regulation problem and the general nonlinear disturbance decoupling problem are stated in [84] and [142], respectively (see also [143]). The general solutions to the nonlinear disturbance decoupling and output regulation problems are, however, highly complex since they require symbolic computation, and they are therefore not well-suited for the automatic online control synthesis,

which is a requirement in this monograph. Here, special solutions are developed that are well-suited for autonomous implementation.

Finally, the question arises whether the fault-hiding approach $\Sigma_{Cr} = (\Sigma_R, \Sigma_C)$ negatively affects the solvability of the reconfiguration problem when compared to the general problem of finding an arbitrary reconfigured controller $\Sigma_{Cr}$.

**Definition 3.11 (Universality of reconfiguration blocks).** The fault-hiding approach with reconfiguration block $\Sigma_R$ to control reconfiguration is called *universal* if the solvability of Problem 3.1 (Problem 3.2) implies the solvability of Problem 3.3 (Problem 3.5).

By definition, universality is defined with respect to particular control reconfiguration problems. In addition, certain universality properties will be stated with respect to particular structures of the reconfigured controller $\Sigma_{Cr}$, and they will be given distinct names. More precise definitions for universality are given below when needed.

The following chapter recalls linear solutions to the problems stated in this chapter. In the linear case, the fault-hiding approach provides a reconfiguration solution whenever it exists. In other words, the fault-hiding approach is universal for linear systems combined with linear control schemes. The further chapters will generalise the fault-hiding solutions from linear systems to two special classes of nonlinear dynamical systems: Hammerstein-Wiener systems and piecewise affine systems. The universality of the fault-hiding approach carries over to the case of Hammerstein-Wiener systems for the cases of exclusive actuator faults and exclusive sensor faults. The idea of virtual sensors (3.9) and virtual actuators (3.10) shown in Fig. 3.10 will be extended by feedthrough of the measured output $\boldsymbol{y}_f$ to the nominal controller in Part III, without changing the the basic idea and the basic properties described in simpler form in this chapter. However, the added feedthrough introduces additional feedback loops into the Figures 3.11 and 3.12.

# Chapter 4
# Linear Reconfiguration Solutions Based on the Fault-Hiding Approach

**Abstract.** This chapter presents linear solutions to the reconfiguration problems defined in Chapter 3 based on the fault-hiding principle. First, nominal linear systems are defined, some of their basic properties are reviewed, and the nominal closed-loop system is defined. It is shown how faults are modelled in linear systems. The linear virtual sensor and the linear virtual actuator are defined, and it is described how they must be designed such that the previously defined reconfiguration goals are satisfied. The solution procedures for obtaining the suitable parameters in the reconfiguration blocks are sketched.

## 4.1 Nominal Linear Systems

Linear systems are viewed as the simplest dynamical approximation of nonlinear systems. They are defined as follows.

**Definition 4.1 (Linear dynamical system [117, 211]).** A *linear time-invariant (LTI) dynamical system* is a system of linear first-order ODEs

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t) \end{cases} \tag{4.1}$$

where all signals are in accordance with Definition 3.2, and where $\boldsymbol{A}$ is the system matrix, $\boldsymbol{B}$ is the input matrix, $\boldsymbol{B}_{\boldsymbol{d}}$ the disturbance input matrix, $\boldsymbol{C}$ is the measurement matrix, and $\boldsymbol{C}_z$ is the relevant output matrix. All matrices are of compatible dimensions, and

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{4.2}$$

is the initial condition. ◇

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 55–86.
springerlink.com

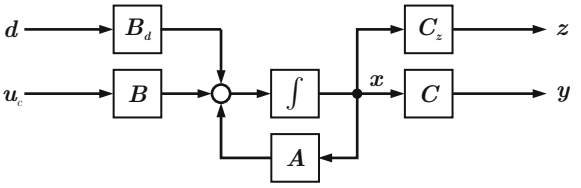


**Fig. 4.1** Linear dynamical system.

A linear system (Fig. 4.1) is the simplest dynamical approximation of the nonlinear system (3.2) that is obtained by linearisation about an operating point $(\boldsymbol{x}_o, \boldsymbol{d}_o, \boldsymbol{u}_o, \boldsymbol{y}_o, \boldsymbol{z}_o)$:

$$\boldsymbol{A} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{u}_c, \boldsymbol{d})\Big|_{\boldsymbol{x}=\boldsymbol{x}_o, \boldsymbol{u}_c=\boldsymbol{u}_o, \boldsymbol{d}=\boldsymbol{d}_o}, \qquad \boldsymbol{B} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{u}_c}(\boldsymbol{x}, \boldsymbol{u}_c, \boldsymbol{d})\Big|_{\boldsymbol{x}=\boldsymbol{x}_o, \boldsymbol{u}_c=\boldsymbol{u}_o, \boldsymbol{d}=\boldsymbol{d}_o} \tag{4.3}$$

$$\boldsymbol{C} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}(\boldsymbol{x})\Big|_{\boldsymbol{x}=\boldsymbol{x}_o}, \qquad \boldsymbol{C}_z = \frac{\partial \boldsymbol{h}_z}{\partial \boldsymbol{x}}(\boldsymbol{x})\Big|_{\boldsymbol{x}=\boldsymbol{x}_o}. \tag{4.4}$$

**Property 4.1 (Stability, ISS, and convergence of linear systems).** *The LTI system (4.1) is 0-GES iff there exist solutions $\boldsymbol{X} = \boldsymbol{X}^T > 0$ and $\boldsymbol{Y} = \boldsymbol{Y}^T > 0$ of the LMI*

$$\boldsymbol{X}\boldsymbol{A} + \boldsymbol{A}^T\boldsymbol{X} + \boldsymbol{Y} < 0. \tag{4.5}$$

*Furthermore, the system (4.1) is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{d})$ and globally exponentially convergent.*

The inequality (4.5) is well-known as the Lyapunov inequality. The convergence property follows immediately from Theorem 2.4.

The $H_\infty$-norm of a linear system has the following alternative characterisations that will be used in this monograph.

**Theorem 4.1 (Characterisation of $H_\infty$-norm [29, 187, 211]).** *Consider the linear dynamical system with throughput*

$$\Sigma : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}_c(t) \end{cases} \tag{4.6}$$

*and with the transfer function*

$$\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s) = \boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{D}, \tag{4.7}$$

*and let $\gamma > 0$ be a scalar. Then,*

$$\|\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s)\|_{H_\infty} = \sup_{s \in \mathbb{C}^+} \mu_{max}(\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s)),$$

*where $\mu_{max}(\cdot)$ is the largest singular value of its argument, and furthermore the following statements are equivalent:*

1. $\|\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s)\|_{H_\infty} < \gamma$.
2. *There exists a feasible solution* $\boldsymbol{X} = \boldsymbol{X}^T > 0$ *to the LMI*

$$\left(\begin{array}{c|c|c} \boldsymbol{A}^T\boldsymbol{X} + \boldsymbol{X}\boldsymbol{A} & \boldsymbol{X}\boldsymbol{B} & \boldsymbol{C}^T \\ \hline \star & -\gamma \boldsymbol{I} & \boldsymbol{D}^T \\ \hline \star & \star & -\gamma \boldsymbol{I} \end{array}\right) < 0. \qquad \diamond$$

The linear system (4.1) is called stabilisable if the pair $(\boldsymbol{A}, \boldsymbol{B})$ is stabilisable. Likewise, it is called detectable if the pair $(\boldsymbol{C}, \boldsymbol{A})$ is detectable.

**Example 4.1 (Linearised model of a ship).** *The linearisation of the ship model (1.1)–(1.4), stated here for later use about the operating point* $\bar{\boldsymbol{x}} = (0.1\,0\,0\,0)^T$ *corresponding to slow surge velocity, and ignoring input saturations, leads to a linear model of the form (4.1) with the model elements*

$$\boldsymbol{A} = \begin{pmatrix} -0.2105 & 0 & 0 & 0 \\ 0 & -0.2841 & -0.054 & 0 \\ 0 & -0.3857 & -0.2381 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad \boldsymbol{B} = \begin{pmatrix} 0.0526 & 0.0526 & 0 \\ 0 & 0 & 0 \\ 0.0238 & -0.0238 & 0.2381 \\ 0 & 0 & 0 \end{pmatrix} \tag{4.8}$$

$$\boldsymbol{B_d} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \quad \boldsymbol{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \boldsymbol{C}_z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.9}$$

*The system is completely controllable from the input* $\boldsymbol{u}_c$ *and completely observable from the measurement* $\boldsymbol{y}$. *The matrix* $\boldsymbol{A}$ *has the set of real eigenvalues* $\sigma(\boldsymbol{A}) = \{-0.115, -0.4072, -0.2105, 0\}$. *The eigenvalue at zero corresponds to the integrator dynamics from yaw velocity r to heading* $\psi$. *The upper left triangular block corresponding to the variables v, w, r is asymptotically stable. The* $H_\infty$*-gains of the transfer functions from all inputs to each separate output of the subsystem* $(v, w, r)$ *are calculated to be* $\|\boldsymbol{T}_{\boldsymbol{u}_c \to v}(s)\|_{H_\infty} = 0.35$, $\|\boldsymbol{T}_{\boldsymbol{u}_c \to w}(s)\|_{H_\infty} = 0.28$, *and* $\|\boldsymbol{T}_{\boldsymbol{u}_c \to r}(s)\|_{H_\infty} = 1.46$.

*Note that in the nonlinear model (1.1)–(1.4), the surge, sway and yaw velocities (see Fig. 1.6) are completely coupled by bilinear feedback-interconnections. In the linearisation, the surge velocity v is not influenced by the sway and yaw velocities w and r, which are driven by the velocity in cascade interconnection. The linearised model thus neglects the influence of sway and yaw motions on the surge velocity.*

*To compare the linearised model to the nonlinear model, their solutions corresponding to the initial surge velocity* $v(0) = 2$, *initial sway velocity* $w(0) = 0.1$, *and initial yaw velocity* $r(0) = -0.79$ *are shown in Fig. 4.2. The figure clearly shows the shortcomings of the linear model: the oscillations that appear in the nonlinear model due to the bilinear coupling do not show up in the linear model. Fig. 4.3 compares the resulting ship positions and headings in the earth-fixed reference frame* $(x, y)$ *starting from the initial position* $(x(0), y(0))^T = (0, 0)^T$ *and the initial heading* $22.5°$. *The heading is symbolised by an oriented triangular ship symbol shown every* 5 *seconds. The figure emphasises that the ship modelled by the nonlinear dynamics rotates by nearly* $\pi$ *rad about the yaw axis while slowing down, whereas the linearised ship slows down with significantly less rotations. Furthermore, the surge*

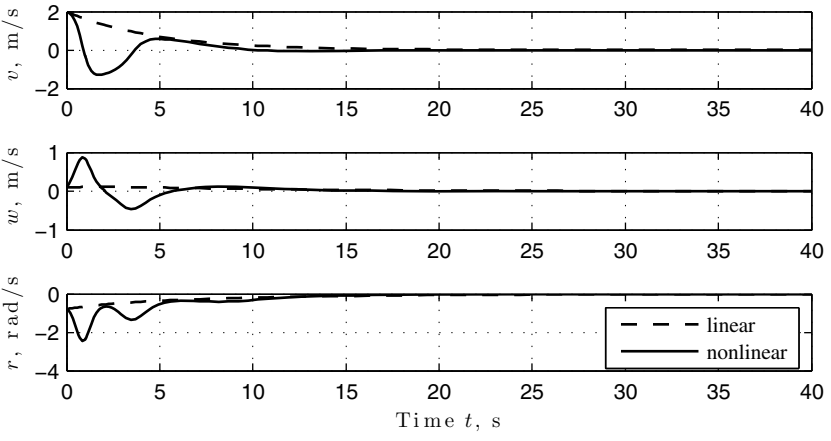


**Fig. 4.2** Comparison of ship responses obtained with nonlinear and linear models in terms of the surge, sway, and yaw velocities.

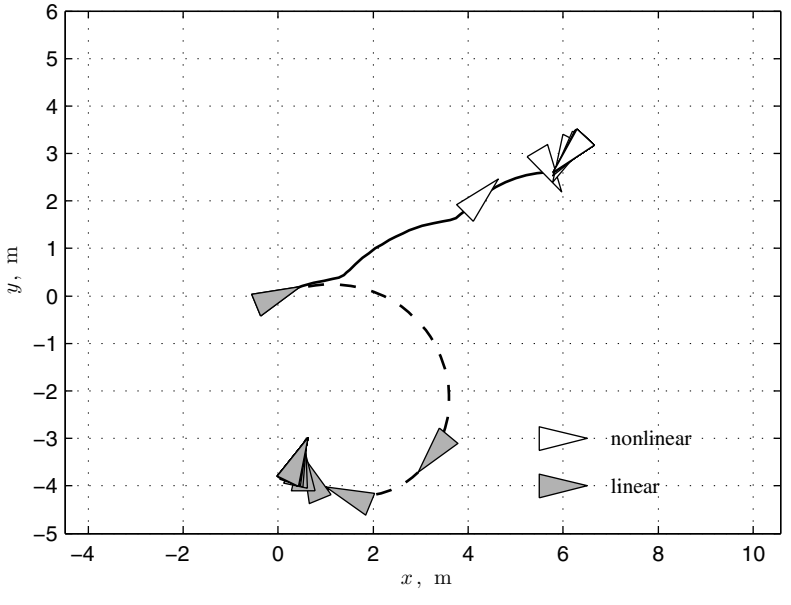


**Fig. 4.3** Comparison of ship responses obtained with nonlinear and linear models in absolute coordinates.

*velocity damping rate is too small, therefore slowing down the ship surge velocity takes longer in the linear ship model than in the nonlinear ship model (see Fig. 4.2), and the linear ship model covers a longer distance than the nonlinear ship model when started from the same initial condition.*

*Finally, it is interesting that for large surge velocities the linearised system is unstable, namely it has an eigenvalue in the right half-plane, in spite of the fact that the physical system and its nonlinear model are globally asymptotically stable in the variables v, w, and r for zero input.*

## 4.2 Nominal Closed-Loop System and Assumption

Consider the linear nominal controller $\Sigma_C$ for the nominal linear plant (4.1),

$$\Sigma_C : \begin{cases} \dot{\boldsymbol{x}}_c(t) &= \boldsymbol{A}_c\boldsymbol{x}_c(t) + \boldsymbol{B}_c\boldsymbol{y}(t) + \boldsymbol{E}_c\boldsymbol{r}(t) \\ \boldsymbol{u}_c(t) &= \boldsymbol{C}_c\boldsymbol{x}_c(t) + \boldsymbol{D}_c\boldsymbol{y}(t) + \boldsymbol{F}_c\boldsymbol{r}(t) \end{cases} \qquad \boldsymbol{x}_c(0) = \boldsymbol{x}_{c0} \tag{4.10}$$

with the internal state $\boldsymbol{x}_c(t) \in \mathbb{R}^{n_c}$ and the reference input $\boldsymbol{r}(t) \in \mathbb{R}^q$. The nominal linear plant (4.1) together with the nominal linear controller (4.10) give rise to the nominal linear closed-loop system $(\Sigma_P, \Sigma_C)$ that satisfies the following assumption.

**Assumption 4.1 (Stabilising and setpoint tracking nominal control).** *The nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$ formed by the nominal linear plant (4.1) and the nominal linear controller (4.10) is internally stable. Furthermore, the tracking error*

$$\boldsymbol{e}_z(t) \triangleq \boldsymbol{r}(t) - \boldsymbol{z}(t) \tag{4.11}$$

*satisfies certain desirable steady-state tracking and performance properties for arbitrary initial conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_{c0}$.*

## 4.3 Faults in Linear Systems

In this section, faults are introduced into the linear system model (4.1).

**Definition 4.2 (Actuator faults in linear systems).** An *actuator fault* in a linear system is an event occurring at time $t_f$ that changes the nominal input matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ to the faulty input matrix $\boldsymbol{B}_f \in \mathbb{R}^{n\times m}$ of the same dimensions. ⋄

**Definition 4.3 (Sensor faults in linear systems).** A *sensor fault* in a linear system is an event occurring at time $t_f$ that changes the nominal measurement matrix $\boldsymbol{C} \in \mathbb{R}^{q\times n}$ to the faulty measurement matrix $\boldsymbol{C}_f \in \mathbb{R}^{q\times n}$ of the same dimensions. ⋄

The above fault definitions include partial component degradation and complete failures, and a single fault may affect more than one component. As an example, an actuator degradation might be modeled by scaling the corresponding input matrix column by factors $\alpha_i$, whereas a complete single actuator failure requires setting the respective column to zero. The same approach works for sensor faults in a similar way:

$$\boldsymbol{B}_f = \begin{pmatrix} \alpha_1 \boldsymbol{b}_1 & \dots & \alpha_m \boldsymbol{b}_m \end{pmatrix}, \qquad \alpha_i \in [0,1],\ i \in \{1,\dots,m\} \tag{4.12}$$

$$\boldsymbol{C}_f = \begin{pmatrix} \beta_1 \boldsymbol{c}_1 \\ \vdots \\ \beta_q \boldsymbol{c}_q \end{pmatrix}, \qquad \beta_i \in [0,1],\ i \in \{1,\dots,q\}. \tag{4.13}$$

In addition to these special cases, the fault definitions 4.2 and 4.3 cover arbitrary changes in the input and measurement matrices. The fault event abruptly changes the nominal system (4.1) to the faulty linear system

$$\Sigma_{Pf} : \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t) \\ \boldsymbol{x}_f(0) &= \boldsymbol{x}_0. \end{cases} \tag{4.14}$$

Up to this point, that fault model can express actuator blockages in so far as they occur at the linearisation point. However, blockage at other points than the operating point may be modelled as disturbances as follows. Let $\bar{u}_j,\ j \in \mathcal{A},\ \mathcal{A} \subset \{1,\dots,m\}$ denote blockage positions of a set of actuators described by an index set $\mathcal{A}$. Indeed, the set $\mathcal{A}$ is a proper subset of the set of all actuators, since the failure of all actuators would result in an autonomous faulty system that cannot be controlled. Thus, the actuator blockage gives rise to the faulty system model

$$\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) + \sum_{j\in\mathcal{A}} \boldsymbol{b}_j \bar{u}_j, \tag{4.15}$$

where $\boldsymbol{B}_f$ is according to Equation (4.12) with $\alpha_j = 0$ for $j \in \mathcal{A}$. In other words, actuators blocked outside the operating point of linearisation act like constant disturbances, whose structure is defined by the corresponding columns of $\boldsymbol{B}$. Strictly speaking, the faulty system with constant disturbance input is an affine system.

Table 4.1 summarises the expressiveness of the linear fault model.

**Table 4.1** Technological faults representable by linear fault models.

| **Technological fault** | **Representable** | **By model parameter** |
|---|---|---|
| Changed actuator gain | ✓ | $\boldsymbol{B}_f$ |
| Changed nonlinear actuator characteristic | × | |
| Changed or reduced actuation range | × | |
| Actuator failure at the operating point | ✓ | $\boldsymbol{B}_f$ |
| Actuator failure off the operating point | (✓) | $\sum_{j\in\mathcal{A}} \boldsymbol{b}_j \bar{u}_j$ (affine) |
| Changed sensor gain | ✓ | $\boldsymbol{C}_f$ |
| Changed nonlinear sensor characteristic | × | |
| Sensor failure | ✓ | $\boldsymbol{C}_f$ |

Legend: ✓: fully representable; (✓): representable leaving the system class; ×: not representable

**Example 4.2 (Linear model of ship subject to faults).** *In the linearised ship model (4.8)–(4.9), faults are introduced as follows. The failure of the yaw rate gyro sensor, fault $f_1$, is modelled by means of a modified output matrix*

$$\boldsymbol{C}_f = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.16}$$

*meaning that the gyro sensor outputs the value zero after its failure. The matrix $\boldsymbol{C}_z$ remains unchanged.*

*A blockage of the rudder $u_3$ in some blockage position $p \in [-1, 1]$ between maximum clockwise rudder moment and maximum counter-clockwise rudder moment, corresponding to the fault $f_2$, is modelled by a modified input matrix*

$$\boldsymbol{B}_f = \begin{pmatrix} 0.0526 & 0.0526 & 0 \\ 0 & 0 & 0 \\ 0.0238 & -0.0238 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{4.17}$$

*whose third column corresponds to the failed rudder, as well as a constant disturbance*

$$\bar{\boldsymbol{d}}_f = \begin{pmatrix} 0 \\ 0 \\ 0.2381 \\ 0 \end{pmatrix} p. \tag{4.18}$$

*The entire faulty system is described by the equations*

$$\begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \bar{\boldsymbol{d}}_f p + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t). \end{cases} \tag{4.19}$$

*The fault $f_3$ meaning a floating rudder is obtained by the special case $p = 0$. The reduced thruster range $f_4$ is not representable within the linear model framework.*

## 4.4 Reconfiguration Problems

The reconfiguration block (3.21) is now a linear system

$$\Sigma_R : \begin{cases} \dot{\boldsymbol{\zeta}}(t) &= \boldsymbol{A}_r\boldsymbol{\zeta}(t) + \boldsymbol{B}_r\boldsymbol{u}_c(t) + \boldsymbol{E}_r\boldsymbol{y}_f(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{C}_r\boldsymbol{\zeta}(t) + \boldsymbol{F}_r\boldsymbol{y}_f(t) \\ \boldsymbol{u}_f(t) &= \boldsymbol{G}_r\boldsymbol{\zeta}(t) + \boldsymbol{H}_r\boldsymbol{u}_c(t), \\ \boldsymbol{\zeta}(0) &= \boldsymbol{\zeta}_0 \end{cases} \tag{4.20}$$

that, together with the linear faulty plant (4.14) and the linear nominal controller (4.10), forms the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_R, \Sigma_C)$. The reconfiguration problems 3.3, 3.4, and 3.5 stated in Chapter 3 are now specified for linear systems, starting with stability.

**Problem 4.1 (Stability recovery for linear systems).** Consider the nominal linear plant $\Sigma_P$ defined in Equation (4.1) and the faulty linear plant $\Sigma_{Pf}$ defined in Equation (4.14). Find a linear reconfiguration block $\Sigma_R$ of the form (4.20) such that

$$\forall \Sigma_C : \{(\Sigma_P, \Sigma_C) \text{ internally stable}\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ internally stable}\}.$$

In other words, every controller that internally stabilises the nominal plant must also stabilise the reconfigured plant.

**Problem 4.2 (Stable asymptotic setpoint tracking recovery for linear systems).** Consider the nominal linear plant $\Sigma_P$ defined in Equation (4.1) and the faulty linear plant $\Sigma_{Pf}$ defined in Equation (4.14). Find a reconfiguration block $\Sigma_R$ of the form (4.20) such that for all nominal controllers $\Sigma_C$ satisfying Assumption 4.1, the reconfigured closed-loop system is internally stable, and

$$\forall \boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t), \boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \lim_{t\to\infty} \Big(z(t) - z_f(t)\Big) = \mathbf{0}. \tag{4.21}$$

In other words, the reconfigured closed-loop system should asymptotically track constant setpoints to the same precision as the nominal closed-loop system.

**Problem 4.3 (Exact stable performance recovery for linear systems).** Consider the nominal plant $\Sigma_P$ defined in Equation (4.1) and the faulty plant $\Sigma_{Pf}$ defined in Equation (4.14). Find a reconfiguration block $\Sigma_R$ of the form (4.20) with the initial condition $\boldsymbol{\zeta}_0$ such that for all nominal controllers $\Sigma_C$ satisfying Assumption 4.1, the reconfigured closed-loop system is stable, and

$$\forall t \geq 0,\ \forall \boldsymbol{r}(t), \boldsymbol{d}(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : z(t) - z_f(t) = \mathbf{0}. \tag{4.22}$$

In other words, for arbitrary reference and disturbance inputs, the reconfigured closed-loop system should respond with exactly the nominal output. This goal is frequently not achievable. The following problem consists in approximating the exact recovery.

**Problem 4.4 (Almost exact stable performance recovery for linear systems).** Consider the nominal plant $\Sigma_P$ defined in Equation (4.1) and the faulty plant $\Sigma_{Pf}$ defined in Equation (4.14). Given a bound $\gamma \in (0, \infty]$ on performance loss, find a

reconfiguration block $\Sigma_R$ of the form (4.20) such that for all nominal controllers $\Sigma_C$ satisfying Assumption 4.1, the reconfigured closed-loop system is stable, and

$$\forall \boldsymbol{u}_c(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \frac{\|z(t) - z_f(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \leq \gamma. \tag{4.23}$$

The latter problem is solvable if the inequality (4.23) is satisfyable for every $\gamma$. The following problem demands a compromise between best-possible output recovery and moderate input power amplification.

**Problem 4.5 (Optimal stable performance recovery for linear systems).** Consider the nominal plant $\Sigma_P$ defined in Equation (4.1) and the faulty plant $\Sigma_{Pf}$ defined in Equation (4.14). Find an optimal reconfiguration block $\Sigma_{R*}$ that, for all nominal controllers $\Sigma_C$ satisfying Assumption 4.1, attains a tradeoff $\lambda\gamma_z + (1-\lambda)\gamma_u$ parameterized over $\lambda \in [0,1]$ between

1. Optimal approximation of the closed-loop output trajectory in the sense that if $z_f^*$ is the output obtained with $\Sigma_{R*}$ and $z_f$ is the output obtained with any other reconfiguration block $\Sigma_R$, then

$$\forall \boldsymbol{u}_c(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \ \gamma_z = \frac{\|z(t) - z_f^*(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \leq \frac{\|z(t) - z_f(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \tag{4.24}$$

   holds, and
2. Minimum amplification of the input signal in the sense that if $\boldsymbol{u}_f^*$ is the control input obtained with $\Sigma_{R*}$ and $\boldsymbol{u}_f$ is the control input obtained with any other reconfiguration block $\Sigma_R$, then

$$\forall \boldsymbol{u}_c(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \ 1 \leq \gamma_u = \frac{\|\boldsymbol{u}_f^*(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \leq \frac{\|\boldsymbol{u}_f(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}}. \tag{4.25}$$

◇

Note that Problem 4.5 reduces to Problem 4.3 whenever the latter is solvable, if total weight is put on output trajectory recovery and zero weight is put on input amplification. Solutions to these problems are next stated separately for the cases of sensor faults and actuator faults.

## 4.5 Linear Virtual Sensor

For the problems with sensor faults ($\boldsymbol{B}_f = \boldsymbol{B}, \boldsymbol{C}_f \neq \boldsymbol{C}$) in the linear faulty plant (4.14), the reconfiguration block $\Sigma_R$ defined in Equation (3.21) is realised by means of a virtual sensor $\Sigma_S$ (Fig. 4.4), which contains an observer for the linear faulty plant (4.14).

**Definition 4.4 (Linear virtual sensor [202]).** The *linear virtual sensor* is the dynamical system

$$\Sigma_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= \boldsymbol{A}_\delta \hat{\boldsymbol{x}}_f(t) + \boldsymbol{B}\boldsymbol{u}_f(t) + \boldsymbol{L}\boldsymbol{y}_f(t) \\ \boldsymbol{u}_f(t) &= \boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{P}\boldsymbol{y}_f(t) + \boldsymbol{C}_\delta \hat{\boldsymbol{x}}_f(t) \end{cases} \tag{4.26}$$

with the initial condition $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$, and where

$$\boldsymbol{A}_\delta \triangleq \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f \tag{4.27}$$

$$\boldsymbol{C}_\delta \triangleq \boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f \tag{4.28}$$

and $\hat{\boldsymbol{x}}_f(t) \in \mathbb{R}^n$ hold. ◇

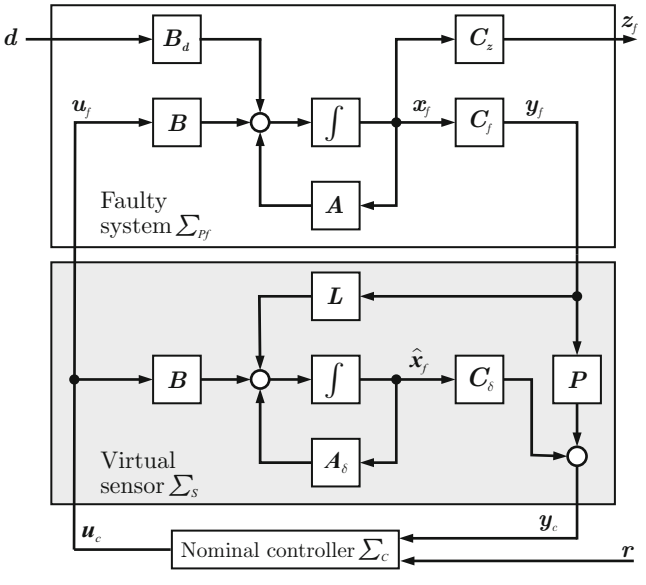


**Fig. 4.4** Reconfigurable control after sensor faults by means of linear virtual sensor.

Note that the linear virtual sensor is a reconfiguration block in accordance with Definition 3.5 that satisfies the inactivity conditions for $\boldsymbol{P} = \boldsymbol{I}$.

The linear reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_S, \Sigma_C)$ is given by the equations (4.14), (4.26), and (4.10). For a better understanding of the following synthesis procedures, it is instructive to study the dynamics of the observation error $\boldsymbol{e}(t) = \hat{\boldsymbol{x}}_f(t) - \boldsymbol{x}_f(t)$ and its impact on the output $\boldsymbol{y}_c$ provided to the controller,

$$\begin{cases} \dot{\boldsymbol{e}}(t) & = \boldsymbol{A}_\delta \boldsymbol{e}(t) - \boldsymbol{B}_{\boldsymbol{d}} \boldsymbol{d}(t) \\ \boldsymbol{y}_\delta(t) & = \boldsymbol{C}_\delta \boldsymbol{e}(t) \\ \boldsymbol{y}_c(t) & = \boldsymbol{C}\boldsymbol{x}_f(t) + \boldsymbol{y}_\delta(t). \end{cases} \tag{4.29}$$

The equation shows that the observation error perturbs the true measurement $\boldsymbol{C}\boldsymbol{x}_f$ through the matrix $\boldsymbol{C}_\delta$. It is straightforward to verify that the Problems 4.2 to 4.5 refer to the reconfigured plant transfer functions $\boldsymbol{T}_{\boldsymbol{u}_c \to z_f}(s)$, $\boldsymbol{T}_{\boldsymbol{d} \to z_f}(s)$, $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}_c}(s)$, and $\boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{y}_c}(s)$. The first three transfer functions are always nominal in the case of sensor faults, namely

$$\begin{aligned} \boldsymbol{T}_{\boldsymbol{u}_c \to z_f}(s) &= \boldsymbol{C}_z (s\boldsymbol{I} - \boldsymbol{A})^{-1} \boldsymbol{B} = \boldsymbol{T}_{\boldsymbol{u}_c \to z}(s) \\ \boldsymbol{T}_{\boldsymbol{d} \to z_f}(s) &= \boldsymbol{C}_z (s\boldsymbol{I} - \boldsymbol{A})^{-1} \boldsymbol{B}_{\boldsymbol{d}} = \boldsymbol{T}_{\boldsymbol{d} \to z}(s) \\ \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}_c}(s) &= \boldsymbol{C} (s\boldsymbol{I} - \boldsymbol{A})^{-1} \boldsymbol{B} = \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s). \end{aligned}$$

The transfer function from the disturbance $\boldsymbol{d}$ to the output $\boldsymbol{y}_c$ is given by

$$\begin{aligned} \boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{y}_c}(s) &= \boldsymbol{C} (s\boldsymbol{I} - \boldsymbol{A})^{-1} \boldsymbol{B}_{\boldsymbol{d}} + \boldsymbol{C}_\delta (s\boldsymbol{I} - \boldsymbol{A}_\delta)^{-1} \boldsymbol{B}_{\boldsymbol{d}} \\ &= \boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{y}}(s) + \boldsymbol{C}_\delta (s\boldsymbol{I} - \boldsymbol{A}_\delta)^{-1} \boldsymbol{B}_{\boldsymbol{d}}, \end{aligned}$$

where the transfer function

$$\boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{y}_\delta}(s) \triangleq \boldsymbol{C}_\delta (s\boldsymbol{I} - \boldsymbol{A}_\delta)^{-1} \boldsymbol{B}_{\boldsymbol{d}} \tag{4.30}$$

represents the effect of the sensor fault visible at the output. Based on Assumption 4.1, the following results about the properties of the reconfigured closed-loop system have been obtained.

**Theorem 4.2 (Reconfigured closed-loop stability recovery [202]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{B}_f = \boldsymbol{B}$, and suppose that Assumption 4.1 holds. Problem 4.1 is solvable by means of a virtual sensor (4.26) if and only if*

$$(\boldsymbol{C}_f, \boldsymbol{A}) \text{ is detectable.} \tag{4.31}$$

*The reconfigured closed-loop system (4.14), (4.10), (4.26) is stable if $\sigma(\boldsymbol{A}_\delta) \subset \mathbb{C}_-$.*

In other words, given nominal closed-loop stability, the reconfigured closed-loop stability depends only on the stability of the error dynamics of the virtual sensor. The latter is determined by the existence of a matrix $\boldsymbol{L}$ such that the matrix $\boldsymbol{A}_\delta = \boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f$ is Hurwitz. Finding a stabilising solution $\boldsymbol{L}$ is a linear observer design problem, which, using duality, is solvable by means of pole-placement algorithms [183, 226] or linear quadratic optimal control algorithms [96, 117].

**Theorem 4.3 (Reconfigured closed-loop tracking recovery [202]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{B}_f = \boldsymbol{B}$, and suppose that Assumption 4.1 holds. Problem 4.2 is solvable by means of a virtual sensor (4.26) in the presence of constant disturbances if and only if Condition (4.31) holds and*

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B_d} \\ \boldsymbol{C}_f & \boldsymbol{0} \end{pmatrix} = \operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B_d} \\ \boldsymbol{C}_f & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} \end{pmatrix}. \tag{4.32}$$

*Given a stabilising gain* $\boldsymbol{L}$*, the gain* $\boldsymbol{P}$ *must be chosen according to*

$$\boldsymbol{P} = \boldsymbol{C}(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f)^{-1}\boldsymbol{B_d}\left(\boldsymbol{C}_f(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f)^{-1}\boldsymbol{B_d}\right)^{\ddagger}.$$

In other words, the output error of the observer is statically decoupled from constant disturbances $\boldsymbol{d}$ by placing a transmission zero in the transfer function $\boldsymbol{T_{d\to y_\delta}}(s) = \boldsymbol{C}_\delta(s\boldsymbol{I} - \boldsymbol{A}_\delta)^{-1}\boldsymbol{B_d}$ at the frequency zero: $\boldsymbol{T_{d\to y_\delta}}(0) = \boldsymbol{0}$.

**Theorem 4.4 (Exact closed-loop performance recovery [180]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with* $\boldsymbol{B}_f = \boldsymbol{B}$*, and suppose that Assumption 4.1 holds. Problem 4.3 is solvable by means of a virtual sensor (4.26) if and only if the infimal detectability subspace* $\mathcal{S}^*_g(\operatorname{im}\boldsymbol{B_d})$ *satisfies the condition*

$$\{\mathcal{S}^*_g(\operatorname{im}\boldsymbol{B_d}) \cap \ker\boldsymbol{C}_f \subseteq \ker\boldsymbol{C}\} \wedge \{(\boldsymbol{C}_f, \boldsymbol{A}) \textit{ is detectable}\}. \tag{4.33}$$

In other words, the disturbance $\boldsymbol{d}$ is dynamically decoupled from the observer output error $\boldsymbol{y}_\delta$ by means of disturbance localisation into a conditioned-invariant detectability subspace $\mathcal{S}^*_g$ that is not visible at $\boldsymbol{y}_\delta$. The transfer function $\boldsymbol{T_{d\to y_\delta}}(s)$ vanishes, namely $\boldsymbol{T_{d\to y_\delta}}(s) = \boldsymbol{0}$. The numerically robust computation of gains $\boldsymbol{L}$ and $\boldsymbol{P}$ that solve Problem 4.3, provided that Condition (4.33) is satisfied, are described in [180].

**Theorem 4.5 (Almost exact closed-loop performance recovery [180]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with* $\boldsymbol{B}_f = \boldsymbol{B}$*, and suppose that Assumption 4.1 holds. Problem 4.4 is solvable by means of a virtual sensor (4.26) if and only if the infimal almost detectability subspace* $\mathcal{S}^*_{b,g}(\operatorname{im}\boldsymbol{B_d})$ *satisfies the condition*

$$\{\mathcal{S}^*_{b,g}(\operatorname{im}\boldsymbol{B_d}) \cap \ker\boldsymbol{C}_f \subseteq \ker\boldsymbol{C}\} \wedge \{(\boldsymbol{C}_f, \boldsymbol{A}) \textit{ is detectable}\}. \tag{4.34}$$

The following theorem shows that the choice of the fault-hiding principle with virtual sensors is not restrictive for the solvability of the reconfiguration problem.

**Theorem 4.6 (Universality of virtual sensors [180]).** *If Problem 3.2 is solvable for sensor faults in an exact sense (in an almost sense), then a solution based on the asymptotic fault-hiding goal and the virtual sensor (4.26) exists in an exact sense (in an almost sense).*

*Proof.* See Appendix D, page 257.

Since Condition (4.55) is frequently not satisfied, a solution to Problem 4.5 is also given that aims at minimising the gain from $\boldsymbol{d}$ to $\boldsymbol{y}_\delta$ in the sense $\mathcal{L}_2 \to \mathcal{L}_2$, which is

given by the $H_\infty$-norm of the transfer function (4.30). Furthermore, the presence of measurement noise $\boldsymbol{n}$ is taken into account, modeled by the modified output equation

$$\boldsymbol{y}_f(t) = \boldsymbol{C}_f \boldsymbol{x}_f(t) + \boldsymbol{n}(t).$$

The noise signals acts uniformly on all components of the measurement vector. The measurement noise has an influence on the output perturbation $\boldsymbol{y}_\delta$ that is characterised by the transfer function

$$\boldsymbol{T}_{\boldsymbol{n}\to\boldsymbol{y}_\delta}(s) = \boldsymbol{C}_\delta(s\boldsymbol{I} - \boldsymbol{A}_\delta)^{-1}\boldsymbol{L} + \boldsymbol{P}. \tag{4.35}$$

**Theorem 4.7 (Optimal closed-loop performance recovery).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{B}_f = \boldsymbol{B}$, and suppose that Assumption 4.1 holds. Problem 4.5 is solvable by means of a linear virtual sensor (4.26) if and only if there exist feasible solutions $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$, $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$, $\boldsymbol{P} \in \mathbb{R}^{q\times q}$, $\gamma_d \in \mathbb{R}$, and $\gamma_n \in \mathbb{R}$ to the convex optimisation problem*

$$\min_{\boldsymbol{X}_s, \boldsymbol{Y}_s, \boldsymbol{P}} \lambda\gamma_d + (1-\lambda)\gamma_n \quad \text{for given } \lambda \in [0,1] \tag{4.36}$$

*subject to*

$$\left(\begin{array}{c|c|c} \boldsymbol{A}^T\boldsymbol{X}_s + \boldsymbol{X}_s\boldsymbol{A} - \boldsymbol{C}_f^T\boldsymbol{Y}_s^T - \boldsymbol{Y}_s\boldsymbol{C}_f & \boldsymbol{X}_s\boldsymbol{B}_{\boldsymbol{d}} & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_d\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_d\boldsymbol{I} \end{array}\right) < 0 \tag{4.37}$$

$$\left(\begin{array}{c|c|c} \boldsymbol{A}^T\boldsymbol{X}_s + \boldsymbol{X}_s\boldsymbol{A} - \boldsymbol{C}_f^T\boldsymbol{Y}_s^T - \boldsymbol{Y}_s\boldsymbol{C}_f & \boldsymbol{Y}_s & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_n\boldsymbol{I} & \boldsymbol{P}^T \\ \hline \star & \star & -\gamma_n\boldsymbol{I} \end{array}\right) < 0 \tag{4.38}$$

$$\boldsymbol{X}_s = \boldsymbol{X}_s^T > 0,\ \gamma_d > 0,\ \gamma_n > 0 \tag{4.39}$$

*The linear virtual sensor gain $\boldsymbol{L}$ is obtained from the equation*

$$\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s, \tag{4.40}$$

*whereas the gain $\boldsymbol{P}$ is directly obtained from the solution of the LMIs (4.37), (4.38).*

*Proof.* By means of elementary operations, the transfer function (4.30) is obtained as a description $\boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{y}_\delta}(s)$ of the influence of the disturbance on the output observation error. Likewise, the transfer function (4.35) is a description $\boldsymbol{T}_{\boldsymbol{n}\to\boldsymbol{y}_\delta}(s)$ of the influence of the disturbance on the output observation error. Using Theorem 4.1 with the substitutions $\boldsymbol{A} \to \boldsymbol{A}_\delta$, $\boldsymbol{B} \to \boldsymbol{B}_{\boldsymbol{d}}$, $\boldsymbol{C} \to \boldsymbol{C}_\delta$, and $\boldsymbol{D} \to \boldsymbol{0}$ implies that the relation $\|\boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{y}_\delta}(s)\|_{H_\infty} \le \gamma_d$ holds if and only if the bilinear matrix inequality

$$\left(\begin{array}{c|c|c} (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f)^T\boldsymbol{X}_{sd} + \boldsymbol{X}_{sd}(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f) & \boldsymbol{X}_{sd}\boldsymbol{B}_{\boldsymbol{d}} & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_d\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_d\boldsymbol{I} \end{array}\right) < 0,\ \boldsymbol{X}_{sd} = \boldsymbol{X}_{sd}^T > 0$$

is feasible, which after the substitution $\boldsymbol{Y}_{sd} \triangleq \boldsymbol{X}_{sd}\boldsymbol{L}$ is equivalent to the LMI

$$\left(\begin{array}{c|c|c} \boldsymbol{A}^T\boldsymbol{X}_{sd} + \boldsymbol{X}_{sd}\boldsymbol{A} - \boldsymbol{C}_f^T\boldsymbol{Y}_{sd}^T - \boldsymbol{Y}_{sd}\boldsymbol{C}_f & \boldsymbol{X}_{sd}\boldsymbol{B_d} & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_d\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_d\boldsymbol{I} \end{array}\right) \prec 0,\ \boldsymbol{X}_{sd} = \boldsymbol{X}_{sd}^T \succ 0.$$

Likewise, the application of the the substitutions $\boldsymbol{A} \to \boldsymbol{A}_\delta$, $\boldsymbol{B} \to \boldsymbol{L}$, $\boldsymbol{C} \to \boldsymbol{C}_\delta$, and $\boldsymbol{D} \to \boldsymbol{P}$ implies that the relation $\|\boldsymbol{T}_{\boldsymbol{n}\to\boldsymbol{y}_\delta}(s)\|_{H_\infty} \leq \gamma_n$ holds if and only if the bilinear matrix inequality

$$\left(\begin{array}{c|c|c} (\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f)^T\boldsymbol{X}_{sn} + \boldsymbol{X}_{sn}(\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C}_f) & \boldsymbol{X}_{sn}\boldsymbol{L} & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_n\boldsymbol{I} & \boldsymbol{P} \\ \hline \star & \star & -\gamma_n\boldsymbol{I} \end{array}\right) \prec 0,\ \boldsymbol{X}_{sn} = \boldsymbol{X}_{sn}^T \succ 0$$

is feasible, which after the substitution $\boldsymbol{Y}_{sn} \triangleq \boldsymbol{X}_{sn}\boldsymbol{L}$ is equivalent to the LMI

$$\left(\begin{array}{c|c|c} \boldsymbol{A}^T\boldsymbol{X}_{sn} + \boldsymbol{X}_{sn}\boldsymbol{A} - \boldsymbol{C}_f^T\boldsymbol{Y}_{sn}^T - \boldsymbol{Y}_{sn}\boldsymbol{C}_f & \boldsymbol{Y}_{sn} & (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)^T \\ \hline \star & -\gamma_n\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_n\boldsymbol{I} \end{array}\right) \prec 0,\ \boldsymbol{X}_{sn} = \boldsymbol{X}_{sn}^T \succ 0.$$

An optimal compromise is obtained by minimising the objective function $\lambda\gamma_d + (1 - \lambda)\gamma_n$ over the feasible solutions of both LMIs. Observing that the recovery of the observer gain $\boldsymbol{L} = \boldsymbol{X}_{sd}^{-1}\boldsymbol{Y}_{sd}$ and $\boldsymbol{L} = \boldsymbol{X}_{sn}^{-1}\boldsymbol{Y}_{sn}$ must be consistent, the unification $\boldsymbol{X}_s \triangleq \boldsymbol{X}_{sd} = \boldsymbol{X}_{sn}$ and $\boldsymbol{Y}_s \triangleq \boldsymbol{Y}_{sd} = \boldsymbol{Y}_{sn}$ results in the LMIs (4.37), (4.38), which completes the proof. ■

The LMI (4.37) is an equivalent characterisation of the $H_\infty$-norm of the transfer function (4.30) from $\boldsymbol{d}$ to $\boldsymbol{y}_\delta$, whereas the LMI (4.38) is an equivalent characterisation of the $H_\infty$-norm of the transfer function (4.35) from $\boldsymbol{n}$ to $\boldsymbol{y}_\delta$. The compromise between the suppression of the disturbance influence and the suppression of measurement noise on the error of the reconstructed output is parameterised by means of the weight $\lambda \in [0, 1]$. The multi-objective optimisation problem (4.36) gives rise to Pareto-optimal solutions, meaning that an improvement with respect to one of the goals cannot be attained without a drawback on the other goal.

**Example 4.3 (Linear virtual sensor for the ship).** *The linearised ship model with the parameters (4.8), (4.9) is now considered for a failure of the gyro sensor ($f_1$). A reconfigurability analysis reveals which of the reconfiguration goals are achievable after the gyro failure. The pair $(\boldsymbol{C}_f, \boldsymbol{A})$ remains observable, therefore also detectable, where $\boldsymbol{C}_f$ has been defined in Equation (4.16) on page 61. Thus, Problem 4.1 is solvable by Theorem 4.2.*

*The Condition (4.32) for setpoint tracking recoverability evaluates to the ranks*

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B_d} \\ \boldsymbol{C}_f & \boldsymbol{0} \end{pmatrix} = 6 = \operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B_d} \\ \boldsymbol{C}_f & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{0} \end{pmatrix},$$

*therefore setpoint tracking recovery in the presence of constant wind is solvable by Theorem 4.3. This result coincides with the intuition that after a failed gyro, the influence of constant wind on the ship is visible in the remaining functional velocity and heading measurements.*

*The relevant spaces in Condition (4.33) for exact performance recoverability are*

$$\mathcal{S}_g^*(\mathrm{im}\,\boldsymbol{B_d}) = \mathbb{R}^4,\ \ker \boldsymbol{C}_f = \mathrm{span}\left(\begin{pmatrix}0\\1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\\0\end{pmatrix}\right),\ \ker \boldsymbol{C} = \mathrm{span}\left(\begin{pmatrix}0\\1\\0\\0\end{pmatrix}\right).$$

*It is easy to see that Condition (4.33) is not satisfied, therefore Theorem 4.4 states that exact performance recovery in the presence of arbitrary wind is impossible. This result is also intuitively clear, since arbitrary time-varying two-dimensional wind has an immediate effect on the acceleration and yaw rate of the ship that becomes visible in the velocity and heading only after one integration.*

*The solvability of optimal performance recovery according to Theorem 4.7 is illustrated in Fig. 4.5. In the figure, the achievable compromise in terms of $\gamma_d, \gamma_n$ is shown for various $\boldsymbol{C}_f = \left(\boldsymbol{c}_1^T\ \beta\boldsymbol{c}_2^T\ \boldsymbol{c}_3^T\right)^T, \beta = 0, 0.2, 0.4, 0.6, 0.8$ and different compromise parameters $\lambda$. The parameter lambda increases from zero (bottom right) to one (top left), as the arrow indicates. The figure shows that the estimation can be arbitrarily well decoupled from the disturbance $\boldsymbol{d}$ at the cost of increased amplification of measurement noise. For complete sensor failure, the previous result of impossible exact recovery is expressed in the non-zero gain $\gamma_d$ for $\beta = 0$.*

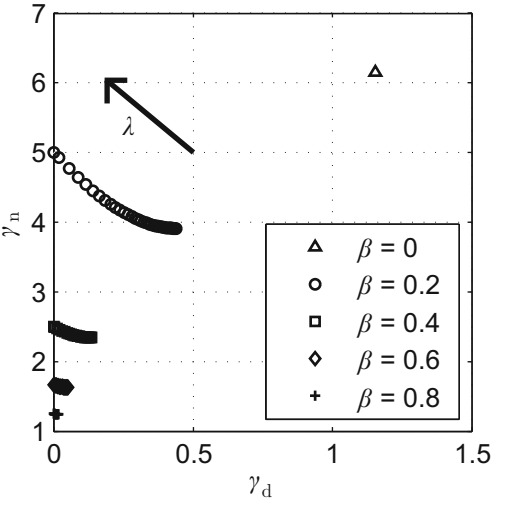


**Fig. 4.5** Reconfigurability analysis for nonlinear ship subject to gyro sensor degradation of varying degree. Points in the lower left corner indicate best reconfigurabiliy, points in the upper right corner indicate worst reconfigurability.

*The response of the nonlinear ship model subject to abrupt gyro sensor failure at* $t_f = 20$ *s is shown in Fig. 4.6. The controller was reconfigured by inserting a linear virtual sensor (4.26) into the closed-loop system, which was designed for setpoint tracking recovery using Theorem 4.3. The virtual sensor well recovers estimates for the yaw rate r as shown in Fig. 4.6, where the actual signals are shown in solid and the estimates are dashed with dot markers. The initial estimation error visible in r quickly settles to zero, in spite of constant wind* $(a_x, a_y) = (0.01, 0.01)$ *in the global reference frame* $(x, y)$ *that translates into the non-constant wind* $(a_v, a_w)$ *in the local reference frame* $(v, w)$ *that is shown in the bottom axis of Fig. 4.6. The wind sets in at time* $t = 30$ *s, as the letter d in the top axis shows.*

*Fig. 4.7 shows the motion of the ship with reconfigured controller, which is capable of avoiding the obstacle and return to its old course due to the reconfiguration.*

Under certain conditions, the exact performance recovery problem after sensor faults can also be solved based on a simplification of the full dynamical linear virtual sensor (4.26) to a static virtual sensor

$$\boldsymbol{y}_c(t) = \boldsymbol{P}\boldsymbol{y}_f(t) \tag{4.41}$$

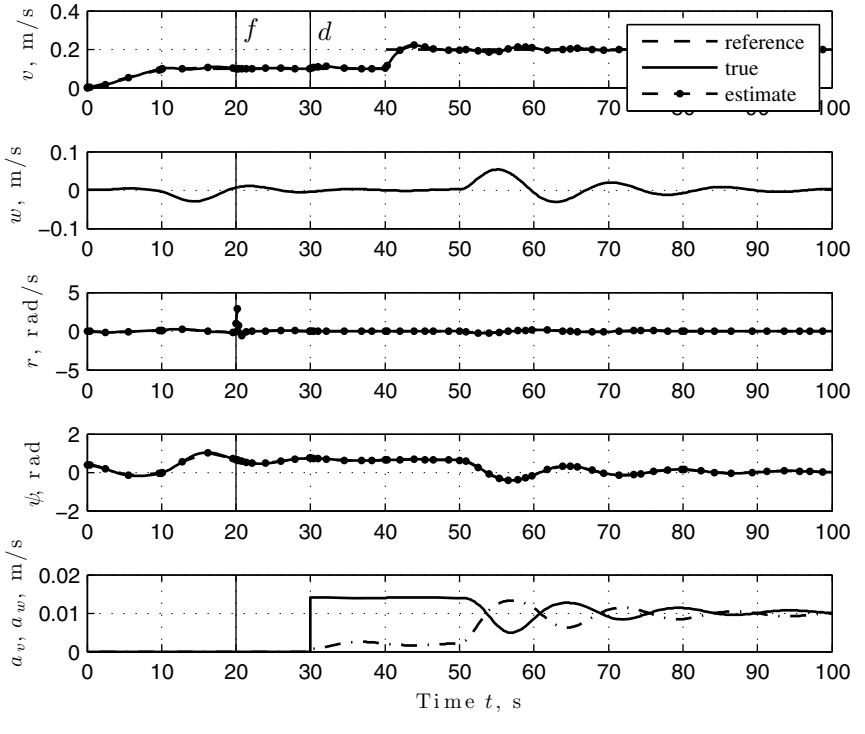


**Fig. 4.6** Response of nonlinear ship subject to gyro sensor failure and control reconfiguration.

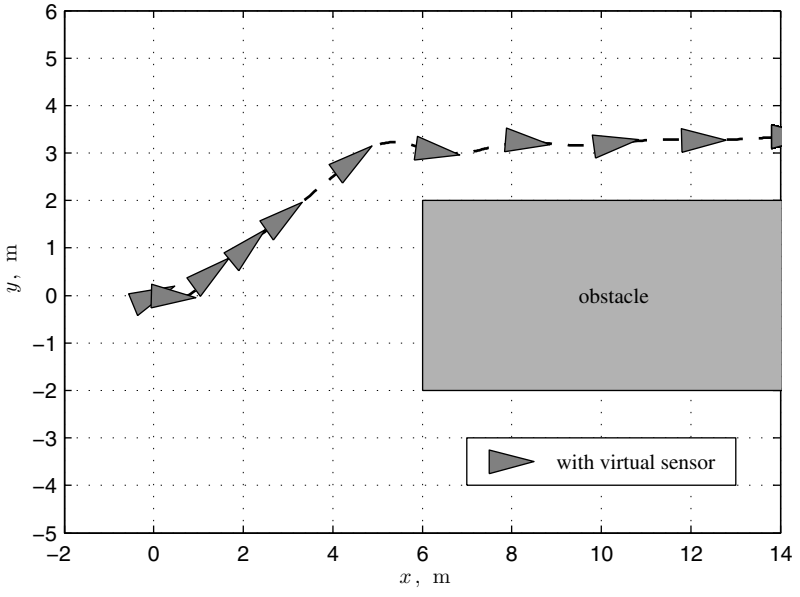


**Fig. 4.7** Motion of nonlinear ship subject to gyro sensor failure and control reconfiguration.

that is placed between the measured output $\boldsymbol{y}_f$ of the faulty plant and the nominal controller. The recoverability of nominal closed-loop performance is characterised in terms of the Markov parameters by the following theorem, which uses the controllability matrices

$$\boldsymbol{S}_C = \left(\boldsymbol{B}\ \boldsymbol{AB}\ \ldots\ \boldsymbol{A}^{n-1}\boldsymbol{B}\right) \tag{4.42}$$

$$\boldsymbol{S}_{C,d} = \left(\boldsymbol{B_d}\ \boldsymbol{AB_d}\ \ldots\ \boldsymbol{A}^{n-1}\boldsymbol{B_d}\right). \tag{4.43}$$

**Theorem 4.8 (Markov-parameter-based exact closed-loop performance recovery using static block [171]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{B}_f = \boldsymbol{B}$, and suppose that Assumption 4.1 holds. Problem 4.3 is solvable by means of a static virtual sensor (4.41) with respect to reference tracking if and only if the condition*

$$\operatorname{rank}\left(\boldsymbol{C}_f\boldsymbol{S}_C\right) = \operatorname{rank}\begin{pmatrix}\boldsymbol{C}_f\boldsymbol{S}_C\\ \boldsymbol{C}\boldsymbol{S}_C\end{pmatrix} \tag{4.44}$$

*is satisfied, where the corresponding gain is*

$$\boldsymbol{P} = \boldsymbol{C}\boldsymbol{S}_C\left(\boldsymbol{C}_f\boldsymbol{S}_C\right)^{+}. \tag{4.45}$$

*It is solvable with respect to disturbance rejection if and only if the condition*

$$\operatorname{rank}\left(\boldsymbol{C}_f\boldsymbol{S}_{C,d}\right) = \operatorname{rank}\begin{pmatrix}\boldsymbol{C}_f\boldsymbol{S}_{C,d}\\ \boldsymbol{C}\boldsymbol{S}_{C,d}\end{pmatrix} \tag{4.46}$$

*is satisfied, where the corresponding gain is*

$$\boldsymbol{P} = \boldsymbol{C}\boldsymbol{S}_{C,d}\left(\boldsymbol{C}_f\boldsymbol{S}_{C,d}\right)^{+}. \tag{4.47}$$

*Problem 4.3 is solvable by means of a static virtual sensor (4.41) with respect to both reference tracking and disturbance rejection if and only if the conditions (4.44) and (4.46) are satisfied at the same time. The corresponding gain is*

$$\boldsymbol{P} = \boldsymbol{C}\left(\boldsymbol{S}_C\ \boldsymbol{S}_{C,d}\right)\left(\boldsymbol{C}_f\left(\boldsymbol{S}_C\ \boldsymbol{S}_{C,d}\right)\right)^{+}. \tag{4.48}$$

## 4.6 Linear Virtual Actuator

For the dual case of actuator faults ($\boldsymbol{B}_f \neq \boldsymbol{B}$, $\boldsymbol{C}_f = \boldsymbol{C}$) in the linear faulty plant (4.14), the reconfiguration block $\Sigma_R$ defined in Equation (3.21) is realised by means of a virtual actuator $\Sigma_A$ which combines the reference model (3.36) and an open-loop observer for the faulty system in a single system. The nonlinear virtual actuator defined in Equation (3.34) becomes the linear virtual actuator (Fig. 4.8).

**Definition 4.5 (Linear virtual actuator [202]).** The *linear virtual actuator* is the dynamical system

$$\Sigma_A : \begin{cases}\dot{\boldsymbol{x}}_{\Delta}(t) &= \boldsymbol{A}_{\Delta}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{B}_{\Delta}\boldsymbol{u}_c(t),\ \boldsymbol{x}_{\Delta}(0) = \boldsymbol{0}\\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t)\\ \boldsymbol{y}_c(t) &= \boldsymbol{y}_f(t) + \boldsymbol{C}\boldsymbol{x}_{\Delta}(t)\end{cases} \tag{4.49}$$

where

$$\boldsymbol{A}_{\Delta} \triangleq \boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M} \tag{4.50}$$

$$\boldsymbol{B}_{\Delta} \triangleq \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} \tag{4.51}$$

and $\boldsymbol{x}_{\Delta}(t) \in \mathbb{R}^n$ holds. ◇

Note that the linear virtual actuator is a reconfiguration block in accordance with Definition 3.5 that satisfies the inactivity conditions for $\boldsymbol{M} = \boldsymbol{0}$ and $\boldsymbol{N} = \boldsymbol{I}$. Indeed, it is the dual system of the linear virtual sensor.

The linear reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_A, \Sigma_C)$ is given by the equations (4.14), (4.49), and (4.10). It is straightforward to verify that the Problems 4.2 to 4.5 refer to the reconfigured plant transfer functions $\boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{z}_f}(s)$, $\boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{z}_f}(s)$,

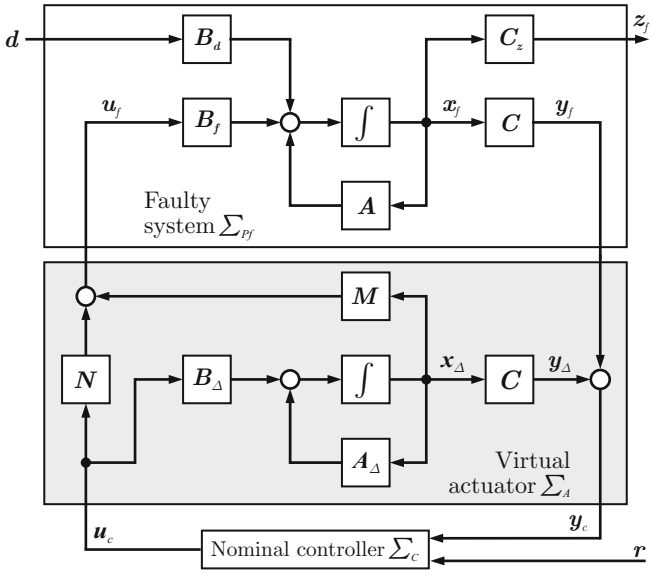


**Fig. 4.8** Reconfigurable control after actuator faults by means of linear virtual actuator.

$\boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{y}_c}(s)$, and $\boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{y}_c}(s)$. The latter three transfer functions are always nominal in the case of actuator faults, namely

$$\begin{aligned}\boldsymbol{T}_{\boldsymbol{d}\to z_f}(s) &= \boldsymbol{C}_z(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B}_{\boldsymbol{d}} = \boldsymbol{T}_{\boldsymbol{d}\to z}(s)\\ \boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{y}_c}(s) &= \boldsymbol{C}(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B} = \boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{y}}(s)\\ \boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{y}_c}(s) &= \boldsymbol{C}(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B}_{\boldsymbol{d}} = \boldsymbol{T}_{\boldsymbol{d}\to\boldsymbol{y}}(s).\end{aligned}$$

The only problem arises from the transfer function $\boldsymbol{T}_{\boldsymbol{u}_c\to z_f}(s) = \boldsymbol{C}_z(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B} + \boldsymbol{C}_z(s\boldsymbol{I}-\boldsymbol{A}_\Delta)^{-1}\boldsymbol{B}_\Delta$, which differs from the nominal transfer function $\boldsymbol{T}_{\boldsymbol{u}_c\to z}(s) = \boldsymbol{C}_z(s\boldsymbol{I}-\boldsymbol{A})^{-1}\boldsymbol{B}$ by the transfer function

$$\boldsymbol{T}_{\boldsymbol{u}_c\to z_\Delta}(s) \triangleq \boldsymbol{C}_z(s\boldsymbol{I}-\boldsymbol{A}_\Delta)^{-1}\boldsymbol{B}_\Delta. \tag{4.52}$$

Based on Assumption 4.1, the following results about the properties of the reconfigured closed-loop system have been obtained.

**Theorem 4.9 (Reconfigured closed-loop stability recovery [202]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds. Problem 4.1 is solvable by means of a linear virtual actuator (4.49) if and only if*

$$(\boldsymbol{A}, \boldsymbol{B}_f) \text{ is stabilisable.} \tag{4.53}$$

*The reconfigured closed-loop system (4.14), (4.10), (4.49) is stable if $\sigma(\boldsymbol{A}_\Delta) \subset \mathbb{C}_-$.*

In other words, given nominal closed-loop stability, the reconfigured closed-loop stability depends only on the stability of the virtual actuator. The latter is determined by the existence of a matrix $\boldsymbol{M}$ such that the matrix $\boldsymbol{A}_{\Delta} = \boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M}$ is Hurwitz. A stabilising solution $\boldsymbol{M}$ can thus be found using pole-placement algorithms [183, 226] or linear quadratic optimal control algorithms [96, 117].

**Theorem 4.10 (Reconfigured closed-loop setpoint tracking recovery [202]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds. Problem 4.2 is solvable by means of a linear virtual actuator (4.49) if and only if Condition (4.53) holds and*

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}_f \\ \boldsymbol{C}_z & \boldsymbol{0} \end{pmatrix} = \operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}_f & \boldsymbol{B} \\ \boldsymbol{C}_z & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}. \tag{4.54}$$

*Given a stabilising gain $\boldsymbol{M}$, the gain $\boldsymbol{N}$ must be chosen according to*

$$\boldsymbol{N} = \left(\boldsymbol{C}_z(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}_f\right)^{\dagger}\boldsymbol{C}_z(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}.$$

In other words, a transmission zero is placed in the transfer function (4.52) at the frequency zero: $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_{\Delta}}(0) = \boldsymbol{0}$.

**Theorem 4.11 (Exact closed-loop performance recovery [125, 180, 202]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds. Problem 4.3 is solvable by means of a linear virtual actuator (4.49) if and only if the supremal stabilisability subspace $\mathcal{V}_g^*(\ker \boldsymbol{C}_z) \subseteq \mathbb{R}^n$ satisfies the condition*

$$\operatorname{im}\boldsymbol{B} \subseteq \mathcal{V}_g^*(\ker \boldsymbol{C}_z) + \operatorname{im}\boldsymbol{B}_f. \tag{4.55}$$

In other words, Condition (4.55) ensures that the fault effect can be contained in a controlled-invariant stabilisability subspace $\mathcal{V}_g^*$ of the state space, which is not observable from the output $\boldsymbol{z}$. Consequently, the transfer function (4.52) vanishes, namely $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_{\Delta}}(s) = \boldsymbol{0}$. The numerically robust computation of gains $\boldsymbol{M}$ and $\boldsymbol{N}$ that solve Problem 4.3, provided that Condition (4.55) is satisfied, is described in [180]. Theorem 4.11 completes the previous results of [125, 202], since it permits the analysis of the existence of stabilising exact decoupling solutions missing in [125, 202], and it implies that the solution is complete, namely a stable performance-recovering solution is found whenever it exists.

**Theorem 4.12 (Almost trajectory recovery with stability [180]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds. Problem 4.4 is solvable by means of a linear virtual actuator (4.49) if and only if the supremal almost stabilisability subspace $\mathcal{V}_{b,g}^*(\ker \boldsymbol{C}_z) \subseteq \mathbb{R}^n$ satisfies the condition*

$$\{\operatorname{im}\boldsymbol{B} \subseteq \mathcal{V}_{b,g}^*(\ker \boldsymbol{C}_z) + \operatorname{im}\boldsymbol{B}_f\} \wedge \{(\boldsymbol{A}, \boldsymbol{B}_f) \textit{ stabilizable}\}. \tag{4.56}$$

Note that the external dynamics relative to $\mathcal{V}^*_{b,g}(\ker \boldsymbol{C}_z)$ always need to be stabilized in this case (see also the proof given in [180]). Since $\mathcal{V}^*_{b,g} \supseteq \mathcal{V}^*_g$ is always true, the conditions for solving the almost trajectory recovery problem are less strict than the conditions for solving the strict trajectory recovery problem, as expected on the basis of the problem definitions. On the other hand, complete stabilizability of the system is necessary in almost trajectory recovery, whereas the weaker requirement of internal stabilizability with respect to $\mathcal{V}^*_g(\ker \boldsymbol{C}_z)$ holds in the strict case.

The following theorem asserts that the fault-hiding-based virtual actuator is not a restriction for solving the exact performance recovery problem.

**Theorem 4.13 (Universality of virtual actuators [180]).** *If Problem 3.2 is solvable for actuator faults in an exact sense (in an almost sense), then a solution based on the strict fault-hiding goal and the virtual actuator (4.49) exists in an exact sense (in an almost sense).*

*Proof.* See Appendix D, page 257.

Since Condition (4.55) is frequently not satisfied, an optimal performance recovery technique is useful in practice, therefore a solution to Problem 4.5 is also given.

**Theorem 4.14 (Optimal closed-loop performance recovery [173]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds. Problem 4.5 is solvable by means of a linear virtual actuator (4.49) if there exist feasible solutions $\boldsymbol{X}_a \in \mathbb{R}^{n\times n}$, $\boldsymbol{Y}_a \in \mathbb{R}^{m\times n}$, $\boldsymbol{N} \in \mathbb{R}^{m\times m}$, $\gamma_u \in \mathbb{R}$, and $\gamma_z \in \mathbb{R}$ to the optimisation problem*

$$\min_{\boldsymbol{X}_a, \boldsymbol{Y}_a, \boldsymbol{N}} \lambda\gamma_z + (1-\lambda)\gamma_u \quad \text{for given } \lambda \in [0,1] \tag{4.57}$$

*subject to*

$$\left(\begin{array}{c|c|c} \boldsymbol{X}_a\boldsymbol{A}^T + \boldsymbol{A}\boldsymbol{X}_a - \boldsymbol{Y}_a^T\boldsymbol{B}_f^T - \boldsymbol{B}_f\boldsymbol{Y}_a & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{Y}_a^T \\ \hline \star & -\gamma_u\boldsymbol{I} & \boldsymbol{N}^T \\ \hline \star & \star & -\gamma_u\boldsymbol{I} \end{array}\right) < 0 \tag{4.58}$$

$$\left(\begin{array}{c|c|c} \boldsymbol{X}_a\boldsymbol{A}^T + \boldsymbol{A}\boldsymbol{X}_a - \boldsymbol{Y}_a^T\boldsymbol{B}_f^T - \boldsymbol{B}_f\boldsymbol{Y}_a & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{X}_a\boldsymbol{C}_z^T \\ \hline \star & -\gamma_z\boldsymbol{I} & \boldsymbol{0}^T \\ \hline \star & \star & -\gamma_z\boldsymbol{I} \end{array}\right) < 0 \tag{4.59}$$

$$\boldsymbol{X}_a = \boldsymbol{X}_a^T > 0,\ \gamma_u \geq 1,\ \gamma_z > 0. \tag{4.60}$$

*The linear virtual actuator gain $\boldsymbol{M}$ is obtained from the equation*

$$\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}, \tag{4.61}$$

*whereas the gain $\boldsymbol{N}$ is directly obtained from the solution of the LMIs (4.58) and (4.59).*

The compromise between optimal output recovery and minimum input amplification is parameterised over the variable $\lambda \in [0,1]$ that is specified by the system designer, where the case $\lambda = 0$ corresponds to minimum input amplification, and the case $\lambda = 1$ corresponds to minimum performance loss. The multi-objective optimisation problem (4.57) gives rise to Pareto-optimal solutions, meaning that an improvement with respect to one of the goals cannot be attained without a drawback on the other goal.

In summary, necessary and sufficient conditions for the solvability of the problems 4.1–4.3 are available along with procedures for computing the gains of the reconfiguration block in order to solve the corresponding problem whenever it is solvable. The Problem 4.5 is an exception in the sense that sufficient but non-necessary conditions are available. The lack of necessity follows from the unification of the variables $\boldsymbol{X}_a$ and $\boldsymbol{Y}_a$ in the constraints (4.58) and (4.59) of the multi-objective optimisation problem 4.57. Dropping one of the two optimisation objectives recovers the feasibility of the corresponding LMIs as a necessary and sufficient condition.

**Example 4.4 (Linear virtual actuator for the ship).** *The ship model (1.1)–(1.4) has been linearised, providing the model elements (4.8), (4.9). First, a reconfigurability analysis reveals which faults can be recovered from the perspective of linear models. In the case of the rudder failure $f_3$, the pair $(\boldsymbol{A}, \boldsymbol{B}_f)$ remains controllable, therefore also stabilisable, where $\boldsymbol{B}_f$ has been defined in Equation (4.17) on page 61. Thus, Problem 4.1 is solvable by Theorem 4.9.*

*The Condition (4.54) for setpoint tracking recoverability evaluates to the ranks*

$$\operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}_f \\ \boldsymbol{C}_z & \boldsymbol{0} \end{pmatrix} = 6 = \operatorname{rank}\begin{pmatrix} \boldsymbol{A} & \boldsymbol{B}_f & \boldsymbol{B} \\ \boldsymbol{C}_z & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix},$$

*therefore setpoint tracking recovery is solvable by Theorem 4.10. This result coincides with the intuition that after a blocked rudder, control over the yaw motion can be regained by using differential thrust on the left and right thrusters.*

*The relevant spaces in Condition (4.55) for exact performance recoverability are*

$$\operatorname{im}\boldsymbol{B} = \operatorname{span}\left(\begin{pmatrix}1\\0\\0\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\\0\end{pmatrix}\right),\ \mathcal{V}_g^*(\ker \boldsymbol{C}_z) = \operatorname{span}\left(\begin{pmatrix}0\\1\\0\\0\end{pmatrix}\right),\ \operatorname{im}\boldsymbol{B}_f = \operatorname{span}\left(\begin{pmatrix}0.91\\0\\0.41\\0\end{pmatrix}, \begin{pmatrix}-0.41\\0\\0.91\\0\end{pmatrix}\right)$$

*and it can be verified that Condition (4.55) is satisfied. By Theorem 4.10, exact performance recovery is possible. The result means that a virtual actuator exists that contains the remaining fault effect within the sway motion w, were it is not visible through the relevant output $\boldsymbol{z}$.*

*The solvability of optimal performance recovery according to Theorem 4.14 is illustrated in Fig. 4.9. In the figure, the achievable compromise in terms of $\gamma_u, \gamma_z$ is shown for various levels of rudder gain loss $\boldsymbol{B}_f = \begin{pmatrix}\boldsymbol{b}_1 & \boldsymbol{b}_2 & \alpha\boldsymbol{b}_3\end{pmatrix}$,*

$\alpha \in \{0, 0.2, 0.4, 0.6, 0.8\}$ *and different compromise parameters* $\lambda$*. Clearly, increasing loss of rudder effectiveness can only be recovered by means of increasing thruster usage.*

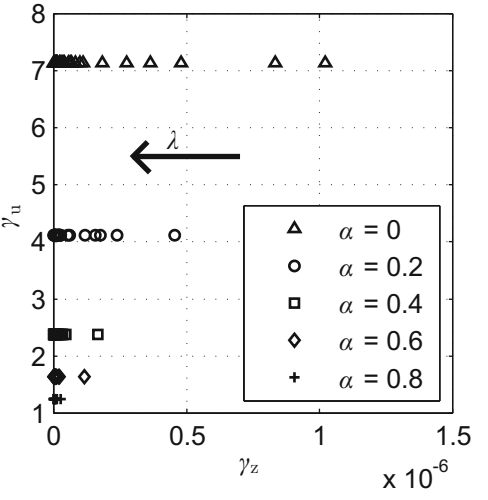


**Fig. 4.9** Reconfigurability analysis for nonlinear ship subject to rudder degradation of varying degree. Points in the lower left corner indicate best reconfigurabiliy, points in the upper right corner indicate worst reconfigurability.

*The response of the nonlinear ship model subject to abrupt rudder failure at* $t_f = 20$ *s is shown in Fig. 4.10. The controller was reconfigured by inserting a linear virtual actuator (4.49) into the closed-loop system, which was designed for exact performance recovery using Theorem 4.11. As expected, the virtual actuator uses the redundancy of differential thrust to manipulate the angular velocity r. However, the input signals* $u_1$*,* $u_2$*, and* $u_3$ *show that the thrust levels* $u_f$ *that the virtual actuator outputs exceed the saturation bounds several times, so that the effective inputs are the cut-off dashed signals shown in the lower three axes of Fig. 4.10. In other words, the linear virtual actuator does not respect the saturations, causes large amplification of the control input, and induces oscillations in surge velocity and yaw rate. Due to the unmodelled saturations, the performance is not exactly recovered in practice, because that requires too large input signals.*

*Fig. 4.11 shows the motion of the ship with reconfigured controller, which is capable of avoiding the obstacle and return to its old course due to the reconfiguration.*

The exact performance recovery problem after actuator faults can sometimes also be solved based on a simplification $\boldsymbol{M} = \boldsymbol{0}$ and $\boldsymbol{y}_c = \boldsymbol{y}_f$ of the full dynamical virtual actuator (4.49) to a static virtual actuator

$$\boldsymbol{u}_f(t) = \boldsymbol{N}\boldsymbol{u}_c(t) \tag{4.62}$$

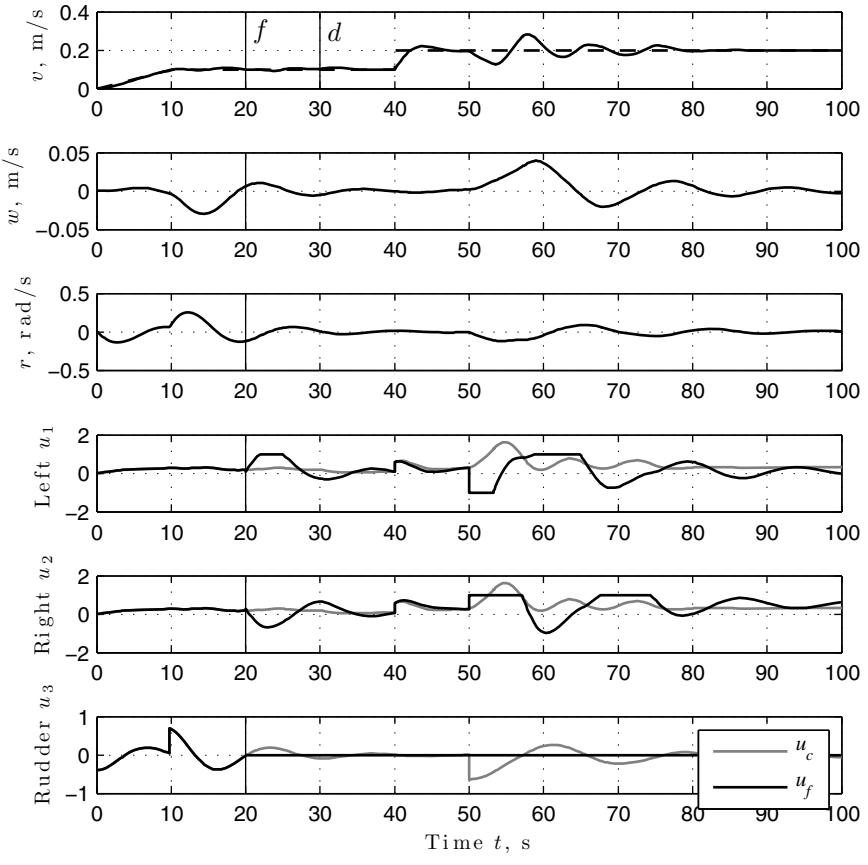


**Fig. 4.10** Response of nonlinear ship subject to rudder failure and control reconfiguration.

that is placed between the controller output $\boldsymbol{u}_c$ and the faulty plant. The recoverability of nominal closed-loop performance is characterised in terms of Markov paremeters by the following theorem, which uses the observability matrix

$$\boldsymbol{S}_O = \begin{pmatrix} \boldsymbol{C} \\ \boldsymbol{CA} \\ \vdots \\ \boldsymbol{CA}^{n-1} \end{pmatrix}. \tag{4.63}$$

**Theorem 4.15 (Markov-parameter-based exact closed-loop performance recovery using static block [171]).** *Consider the nominal linear plant (4.1) and the faulty linear plant (4.14) with $\boldsymbol{C}_f = \boldsymbol{C}$, and suppose that Assumption 4.1 holds.*

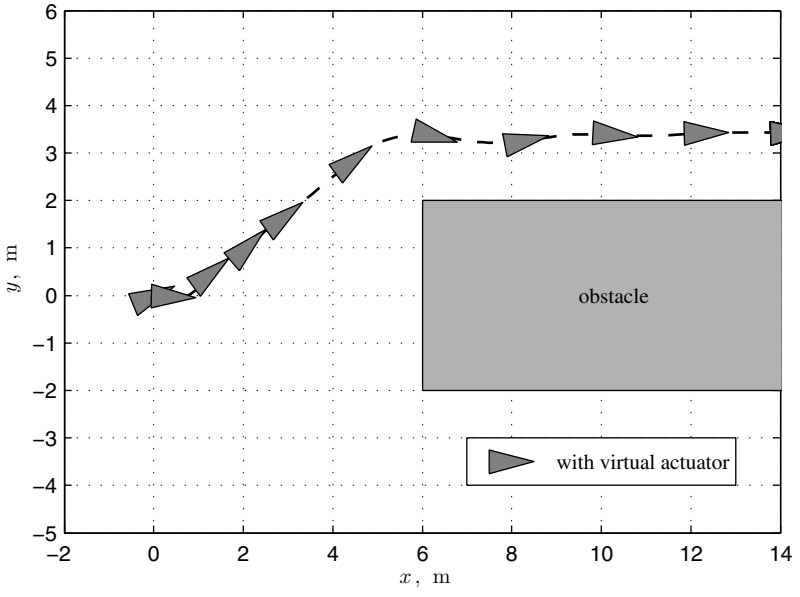


**Fig. 4.11** Motion of nonlinear ship subject to rudder failure and control reconfiguration.

*Problem 4.3 is solvable by means of a static virtual actuator (4.62) if and only if the condition*

$$\operatorname{rank}\left(\boldsymbol{S}_O\boldsymbol{B}_f\right) = \operatorname{rank}\left(\boldsymbol{S}_O\boldsymbol{B}_f\ \boldsymbol{S}_O\boldsymbol{B}\right) \tag{4.64}$$

*is satisfied, where the corresponding gain is*

$$\boldsymbol{N} = \left(\boldsymbol{S}_O\boldsymbol{B}_f\right)^{+}\boldsymbol{S}_O\boldsymbol{B}. \tag{4.65}$$

## 4.7 Combination of Virtual Sensor with Virtual Actuator

The simultaneous occurrence of actuator and sensor faults is now considered. To reconfigure this class of faults, the virtual sensor and virtual actuator are combined,

$$\Sigma_R : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= \boldsymbol{A}_\delta\hat{\boldsymbol{x}}_f(t) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{L}\boldsymbol{y}_f(t) \\ \hat{\boldsymbol{y}}_c(t) &= \boldsymbol{P}\boldsymbol{y}_f(t) + \boldsymbol{C}_\delta\hat{\boldsymbol{x}}_f(t) \\ \dot{\boldsymbol{x}}_\Delta(t) &= \boldsymbol{A}_\Delta\boldsymbol{x}_\Delta(t) + \boldsymbol{B}_\Delta\boldsymbol{u}_c(t) \\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{N}\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) &= \hat{\boldsymbol{y}}_c(t) + \boldsymbol{C}\boldsymbol{x}_\Delta(t), \end{cases} \tag{4.66}$$

with $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f,0}$, $\boldsymbol{x}_\Delta(0) = \boldsymbol{x}_{\Delta 0}$, and where $\boldsymbol{A}_\delta$, $\boldsymbol{C}_\delta$, $\boldsymbol{A}_\Delta$, and $\boldsymbol{B}_\Delta$ are defined in Equations (4.27), (4.28), (4.50), and (4.51), respectively. The combination of the

reconfiguration block (4.66) with the faulty plant (4.14) and a state transformation to the observation error $\boldsymbol{e}(t) = \hat{\boldsymbol{x}}_f(t) - \boldsymbol{x}_f(t)$ and the fictitious state $\tilde{\boldsymbol{x}}(t) = \boldsymbol{x}_f(t) + \boldsymbol{x}_\Delta(t)$ leads to the model of the reconfigured plant

$$\begin{pmatrix} \dot{\tilde{\boldsymbol{x}}}(t) \\ \dot{\boldsymbol{e}}(t) \\ \dot{\boldsymbol{x}}_\Delta(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A}_\delta & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{A}_\Delta \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{e}(t) \\ \boldsymbol{x}_\Delta(t) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{0} \\ \boldsymbol{B}_\Delta \end{pmatrix} \boldsymbol{u}_c(t) + \begin{pmatrix} \boldsymbol{B}_d \\ -\boldsymbol{B}_d \\ \boldsymbol{0} \end{pmatrix} \boldsymbol{d}(t)$$

$$\tilde{\boldsymbol{x}}(0) = \boldsymbol{x}_0 + \boldsymbol{x}_{\Delta 0},\ \boldsymbol{e}(0) = \hat{\boldsymbol{x}}_{f,0} - \boldsymbol{x}_0,\ \boldsymbol{x}_\Delta(0) = \boldsymbol{x}_{\Delta 0}$$

$$\begin{pmatrix} \boldsymbol{y}_f(t) \\ \hat{\boldsymbol{y}}_c(t) \\ \boldsymbol{y}_c(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{C} & \boldsymbol{0} & -\boldsymbol{C}_f \\ \boldsymbol{C} & \boldsymbol{C}_\delta & -\boldsymbol{C} \\ \boldsymbol{C} & \boldsymbol{C}_\delta & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{e}(t) \\ \boldsymbol{x}_\Delta(t) \end{pmatrix}$$

$$\boldsymbol{z}_f(t) = \begin{pmatrix} \boldsymbol{C}_z & \boldsymbol{0} & -\boldsymbol{C}_z \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(t) & \boldsymbol{e}(t) & \boldsymbol{x}_\Delta(t) \end{pmatrix}^T .$$

This model shows that the asymptotic fault-hiding goal is achievable with the initialisations $\hat{\boldsymbol{x}}_{f,0} = \boldsymbol{x}_0$ and $\boldsymbol{x}_{\Delta 0} = \boldsymbol{0}$. Straightforward calculations show that the transfer function relations $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}_c}(s) = \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{y}}(s)$ and $\boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{z}_f}(s) = \boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{z}}(s)$ are satisfied. It remains to recover the transfer functions $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_f}(s)$ and $\boldsymbol{T}_{\boldsymbol{d} \to \boldsymbol{y}_c}(s)$ to their nominal counterparts.

The separate results and conditions regarding stability recovery, setpoint tracking recovery, and performance recovery also apply to the case of combined sensor and actuator faults. Instead of repeating all these results, the interesting case for exact performance recovery is explicitly stated.

**Theorem 4.16 (Exact closed-loop performance recovery after combined sensor and actuator faults [180]).** *Consider the faulty linear system (4.14) with both actuator and sensor faults. Problem 4.3 is solvable iff the infimal detectability subspace* $\mathcal{S}_g^*(\mathrm{im}\,\boldsymbol{B}_d)$ *and the supremal stabilisability subspace* $\mathcal{V}_g^*(\ker \boldsymbol{C}_z) \subseteq \mathbb{R}^n$ *satisfy the Conditions (4.33) and (4.55).*

In other words, trajectory recovery is achievable iff the disturbance decoupled estimation problem and the disturbance decoupling of the output correction problem are both solvable.

*Remark 4.1.* It may appear that the trajectory recovery problem for combined actuator and sensor faults were equivalent to the stable disturbance decoupling problem for measured disturbance (DDPM). The latter problem is solvable iff $\mathcal{S}_g^*(\mathrm{im}\,\boldsymbol{B}_d) \subseteq \mathcal{V}_g^*(\ker \boldsymbol{C}_z)$, which is stricter than the solvability of trajectory recovery after actuator and sensor faults as derived above. The reason is that DDPM refers to a different control structure: The controller uses only output measurements to determine the control input, while the combination of a virtual actuator and a virtual sensor allows more freedom. Indeed, the virtual actuator uses the full state of a difference system prediction to emulate the missing control input that was lost due to the actuator faults. Hence, it is not necessary that $\mathcal{S}_g^* \subseteq \mathcal{V}_g^*$, which is the key to the separation result.

Finally, a global algorithm for coordinated application of the results presented in this chapter is sketched. Since the virtual sensor and the virtual actuator both satisfy the inactivity conditions and therefore qualify as reconfiguration blocks in the sense of Definition 3.5, they can both be implemented according to Equation (4.66) as part of the feedback control loop from the outset by setting them inactive. As soon as a fault is isolated, the fault models $\boldsymbol{B}_f$ and $\boldsymbol{C}_f$ are constructed.

In order to test the recoverability from the sensor faults, the conditions (4.44), (4.46), (4.33), (4.32), and (4.31) are checked in this sequential order in order to find out which is the strongest achievable reconfiguration goal based on the simplest-possible virtual sensor (where static virtual sensors are preferred over dynamical ones). Likewise, to test the recoverability from the actuator faults, the conditions (4.64), (4.55), (4.54), and (4.53) are checked in this sequential order in order to find out which is the strongest achievable reconfiguration goal based on the simplest-possible virtual actuator (again, static virtual actuators being preferred over dynamical ones). The solvability results are typically reported to the plant operators at a higher supervision level.

The gains for the linear virtual sensor and actuator should be computed as solutions of the optimisation problems (4.36) and (4.57), which may be solved in parallel with the solvability tests. This recommendation is due to the fact that excessive control input signal amplitudes are avoidable by careful selection of the weight parameter $\lambda$ exclusively in this method. As soon as the resulting gains are available, the linear virtual sensor and the linear virtual actuator are updated.

## 4.8 Comparison of Virtual Actuator and Dual Observer

In this section the relationship between the dual observer introduced in [112][1] and the virtual actuator introduced in [202] is studied. Both systems distribute control signals from unavailable inputs to available ones. In this sense, they are the opposite of state observers, which reconstruct complete state information from partial measurements. It is shown in this paper that the dual observer is a particular case of the virtual actuator.

In 1962, Rosenbrock emphasised the importance of extending the design of feedback controllers beyond the stabilisation problem in order to influence the input-output (I/O) behavior of the closed-loop system [183]. The perfect pole assignment that he proposed is achievable by means of state-feedback control if the whole system state is measurable (full-state sensing), or by means of output injection control if the controller has access to the innovations of all state variables (full-state actuation). These situations are shown on top of Fig. 4.12 and denoted by state feedback or output injection, respectively.

The fundamental implementation problems of these control structures are the unavailability of all states for measurement on the one hand, and limitations to independently influencing all state innovations on the other hand. The former problem

[1] In [112], Luenberger credits the dual observer to an unpublished manuscript by F. M. Brasch.

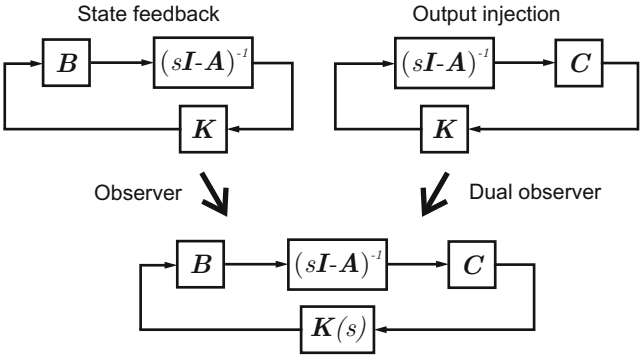


**Fig. 4.12** Dual control realisation problems.

led to the development of the state observer [110, 111], which provides the missing state information and represents a well-interpretable design procedure for obtaining dynamical compensators [31, 32]. The latter problem gave rise to the dual observer [112], which approximates the missing control action by means of the available control channels. In both cases, a static feedback control law $\boldsymbol{K}$ is replaced by a dynamical compensator $\boldsymbol{K}(s)$ (bottom of Fig. 4.12). The introduction of new poles into the feedback loop by the controller $\boldsymbol{K}(s)$ is the price paid for enhancing the design freedom in order to tailor the I/O behavior according to the given closed-loop specifications.

The observer design problem and the role of state observers in the feedback loop have attracted much attention since that time. This attention is partly due to the simplification of the overall compensator synthesis due to the separate design of the observer and the state-feedback controller. However, the dual problem of influencing the innovations of all state variables by means of a dynamical component in the feedback loop has not been thoroughly studied. This is surprising because one could expect an equal interest in both problems due to the well-known duality[2] between observability and controllability, which are the plant properties that enable arbitrary pole assignment. The lack of interest might be explained by the fact that the dual observation problem was perceived as a particular case of the compensation problem. The dual observer is next introduced, starting with the definition of the output injection realisation problem.

**Output injection realisation problem.** Consider the output injection control law

$$\tilde{\boldsymbol{u}}(t) = \tilde{\boldsymbol{L}}(\boldsymbol{y}(t) + \boldsymbol{V}\boldsymbol{r}(t)) \tag{4.67}$$

for the system (4.1) (Fig. 4.13), which yields the closed-loop system (4.1), (4.67)

$$\dot{\boldsymbol{x}}(t) = (\boldsymbol{A} + \tilde{\boldsymbol{L}}\boldsymbol{C})\boldsymbol{x}(t) + \tilde{\boldsymbol{L}}\boldsymbol{V}\boldsymbol{r}(t) \tag{4.68}$$

[2] A broad and detailed treatment of duality is available in [113].

whose transfer function from $\boldsymbol{r}$ to $\boldsymbol{y}$,

$$\boldsymbol{G}_{\boldsymbol{r}\to\boldsymbol{y}}(s) = \boldsymbol{C}\left(s\boldsymbol{I} - (\boldsymbol{A} + \tilde{\boldsymbol{L}}\boldsymbol{C})\right)^{-1}\tilde{\boldsymbol{L}}\boldsymbol{V}, \tag{4.69}$$

has the poles assigned by the appropriate choice of the matrix $\tilde{\boldsymbol{L}}$.

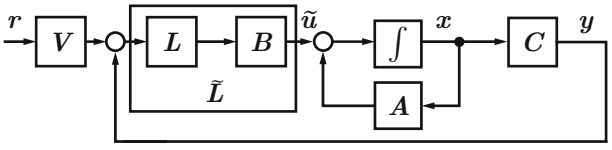


**Fig. 4.13** Output injection realisation problem.

It is known that the eigenvalues of $(\boldsymbol{A} + \tilde{\boldsymbol{L}}\boldsymbol{C})$ can be assigned arbitrary (stable) values by a suitable choice of $\tilde{\boldsymbol{L}}$ if and only if the pair $(\boldsymbol{C}, \boldsymbol{A})$ is observable (detectable) [112, 148]. Output injection control (4.67) is immediately implementable if all state innovations are directly accessible as shown in the upper right diagram of Fig. 4.12, where $\tilde{\boldsymbol{L}}$ is called $\boldsymbol{K}$. Furthermore, the output injection (4.67) is implementable on the plant (4.1), if $\tilde{\boldsymbol{L}}$ can be factorised into $\tilde{\boldsymbol{L}} = \boldsymbol{B}\boldsymbol{L}$, which is possible if and only if the condition

$$\operatorname{im}\tilde{\boldsymbol{L}} \subseteq \operatorname{im}\boldsymbol{B} \tag{4.70}$$

is satisfied (Fig. 4.13). Otherwise, the pole assignment for the transfer function (4.69) can only be approximately realised in the sense that the realisation introduces new, freely assignable poles into the closed-loop system.

**Problem 4.6 (Approximate output feedback realisation [112]).** Consider the system (4.1) the output injection (4.67), and suppose that Condition (4.70) is not satisfied. Realise the output feedback by means of a dynamic controller such that the transfer function $\boldsymbol{G}_{\boldsymbol{r}\to\boldsymbol{y}}(s)$ of the closed-loop system from $\boldsymbol{r}$ to $\boldsymbol{y}$ has the eigenvalues of the matrix $(\boldsymbol{A} + \tilde{\boldsymbol{L}}\boldsymbol{C})$ as dominating poles.

**Dual observer approach.** Problem 4.6 is addressed in [112] and [148] by means of the *dual observer* (Fig. 4.14)

$$\Sigma_{DO}: \begin{cases} \dot{\boldsymbol{s}}(t) &= \boldsymbol{F}\boldsymbol{s}(t) + \boldsymbol{J}\tilde{\boldsymbol{w}}(t) \\ \boldsymbol{u}(t) &= \boldsymbol{M}\boldsymbol{s}(t) + \boldsymbol{N}\tilde{\boldsymbol{w}}(t) \\ \boldsymbol{w}(t) &= \boldsymbol{y}(t) + \boldsymbol{C}\boldsymbol{s}(t) \\ \tilde{\boldsymbol{w}}(t) &= \boldsymbol{w}(t) + \boldsymbol{V}\boldsymbol{r}(t) \end{cases} \tag{4.71}$$

with the initial condition $\boldsymbol{s}(0) = \boldsymbol{z}_0$ subject to the constraints

$$\boldsymbol{F} = \boldsymbol{A} - \boldsymbol{B}\boldsymbol{M}. \tag{4.72}$$

$$\boldsymbol{J} = \tilde{\boldsymbol{L}} - \boldsymbol{B}\boldsymbol{N} \tag{4.73}$$

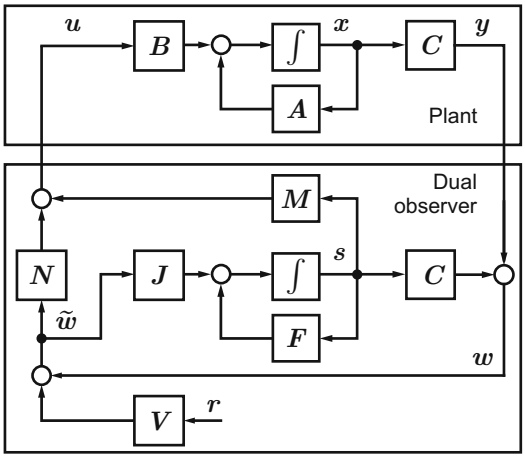


**Fig. 4.14** Dual observer for dynamically realizing output injection.

The dual observer mimics the effect of the missing control input $\tilde{\boldsymbol{u}}$ by using the existing input $\boldsymbol{u}$ (see also [134]).

**Theorem 4.17 (Output feedback realisation by a dual observer [112]).** *The dual observer (4.71) with the constraints (4.72), (4.73) solves Problem 4.6. Furthermore, the constraints (4.72), (4.73) have a solution in $\boldsymbol{M}$ and $\boldsymbol{N}$ such that $\boldsymbol{F}$ has arbitrarily given (stable) eigenvalues, if and only if the system (4.1) is controllable (stabilisable).*

The system (4.71) is called a dual observer in [112] because it solves a problem that is dual to state observation, as illustrated in Fig. 4.12.

*Remark 4.2.* In [112, Th. 5], complete observability is stated as an additional condition for dual observer synthesis (the statement refers to reduced-order dual observers, to be precise). However, Luenberger considered the free pole placement problem for the entire closed-loop system, whereas we consider the matrix $\boldsymbol{A} + \tilde{\boldsymbol{L}}\boldsymbol{C}$ as given.

**Synthesis.** The degrees of freedom lie in the choice of $\boldsymbol{M}$ and $\boldsymbol{N}$. They are used as follows in the design process. The system poles affected by the output feedback are determined by choosing the matrix $\tilde{\boldsymbol{L}}$. Then the dual observer poles can be assigned by choosing $\boldsymbol{F}$, which determines the gain $\boldsymbol{M}$. Finally, it is possible to move the system zeros by changing $\boldsymbol{N}$. Often $\boldsymbol{N} = \boldsymbol{0}$ is chosen.

*Remark 4.3.* The dual observer was originally proposed in [112] as a governor for a linear function $\boldsymbol{s}$ of the plant state $\boldsymbol{x}$ (estimate $\hat{\boldsymbol{x}} = \boldsymbol{T}\hat{\boldsymbol{s}}$). Without loss of generality and for the sake of clarity, this transformation is here assumed to be the identity: $\boldsymbol{T} = \boldsymbol{I}$.

*Remark 4.4.* The last equation of the dual observer (4.71) is different in the original definition in [112], where $\tilde{\boldsymbol{w}} = \boldsymbol{w}$ holds. The reference signal $\boldsymbol{r}$ is added here to emphasise that the eigenvalues added by the dual observer affect the transfer functions $\boldsymbol{G}_{\boldsymbol{r}\to\boldsymbol{y}}(s)$, but are structurally hidden from $\boldsymbol{G}_{\boldsymbol{r}\to\boldsymbol{w}}(s)$.

**Comparison with virtual actuator.** The following corollary to Theorem 4.17 first mentions a special context for the virtual actuator that establishes identity between the dual observer and the virtual actuator.

**Corollary 4.1 (Virtual actuator generalises dual observer).** *The* virtual actuator *used with unity feedback is identical to the* dual observer.

*Proof.* The statement follows from attaching unity feedback $\boldsymbol{u}_c(t) = \boldsymbol{y}_c(t) + \boldsymbol{V}\boldsymbol{r}(t)$ to the virtual actuator (4.49) and comparison to the dual observer (4.71). The closed-loop systems are identical up to signal name exchanges $\boldsymbol{s} \to \boldsymbol{x}_{\Delta}$, $\tilde{\boldsymbol{w}} \to \boldsymbol{u}_c$, $\boldsymbol{u} \to \boldsymbol{u}_f$, $\boldsymbol{w} \to \boldsymbol{y}_c$, and $\boldsymbol{y} \to \boldsymbol{y}_f$, and the matrix variable exchanges $\boldsymbol{B} \to \boldsymbol{B}_f$ and $\tilde{\boldsymbol{L}} \to \boldsymbol{B}$. ∎

Note that, in addition, the virtual actuator (4.49) with attached static linear state-feedback $\boldsymbol{u}(t) = \boldsymbol{K}\boldsymbol{y}(t) + \boldsymbol{V}\boldsymbol{r}(t)$ is equivalent to the system $\dot{\boldsymbol{x}}_{\Delta}(t) = \boldsymbol{F}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{J}\boldsymbol{K}\boldsymbol{y}_c(t) + \boldsymbol{K}^{+}\boldsymbol{V}\boldsymbol{r}(t)$, $\boldsymbol{u}_f(t) = \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{K}\boldsymbol{y}_c(t) + \boldsymbol{K}^{+}\boldsymbol{V}\boldsymbol{r}(t)$, which is identical to the dual observer (4.71). However, such unifying transformations are not possible with dynamical or nonlinear controllers, which highlights the generalisation of the virtual actuator. This generalisation is largely due to the external signals $\boldsymbol{u}_c$ and $\boldsymbol{y}_c$, which are constrained to $\boldsymbol{u}_c = \boldsymbol{y}_c$ in the dual observer.

Although the motivations for dual observer design and virtual actuator design differ, the main differences, apart from the restriction of the dual observer to unity feedback, concern the notation. The desired input matrices that appear in the constraints (4.72) and (4.50) are distinct and denoted by $\tilde{\boldsymbol{L}}$ in output feedback realisation, and by $\boldsymbol{B}$ in control reconfiguration after actuator faults. The given input matrices to the given plants differ, called $\boldsymbol{B}$ in output feedback realisation and $\boldsymbol{B}_f$ in control reconfiguration.

## 4.9 Extensions and Discussion

Further extensions have been worked out both for actuator faults and sensor faults in linear systems. These extensions, which are not described here in detail, concern

- synthesis strategies based on Markov-parameter descriptions of the nominal and faulty system for dynamical virtual actuators [175], and
- reduced-order virtual actuators and virtual sensors [124].

Note that all solvability conditions stated in this chapter depend only on the parameters of the nominal and faulty plants. Therefore, reconfigurability is a pure system property. In particular, it does not depend on the nominal controller. This feature is a consequence of the formulation of the Problems 4.1–4.5 independently of the controller.

Problem 4.5 on optimal performance recovery is by definition always solvable. If the exact performance recovery problem is solvable, then the solutions computed based on the optimisation approaches given in Theorem 4.14 (actuator case) and given in Theorem 4.7 (sensor case) correspond to the exact performance recovering solutions based on geometric theory in Theorem 4.11 (actuator case) and in Theorem 4.4 (sensor case). The geometric conditions are highly useful for analysing a priori which fault scenarios are reconfigurable in what sense, whether or not the parameters for the virtual actuator and virtual sensor are computed based on the same theorem. Provided that LMIs can be efficiently and reliably solved online, the use of the optimisation approaches is an attractive alternative, since the choices of target poles or weight matrices that are necessary for pole placement and linear quadratic synthesis are replaced by the choice of a scalar parameter $\lambda$, which is considerably less complex and therefore easier.

The universality of the linear virtual actuator and the linear virtual sensor has been first shown in [180]. The significance of this result consists in the assertation that choosing the fault-hiding principle is not restrictive. The relation between the virtual actuator and the dual observer has been first discussed in [176].

In summary, this chapter has demonstrated that the theory of reconfigurable control based on fault-hiding approaches can be considered as fairly complete for linear systems, concluding Part I of this monograph. Reconfiguration solutions for the exclusive and combined occurrence of actuator and sensor faults based on the fault-hiding approach are available, and necessary and sufficient solvability conditions for the stability recovery, tracking recovery, and performance recovery problems are known. The case study [177] has revealed two properties of the linear fault-hiding approach in the context of a process control application. First, the fault-hiding principle is in principle practically feasible. Second, the linear approaches have the following significant limitations if the plant exhibits significant nonlinear behavior: linear systems cannot express physical constraints on control inputs, and they cannot represent nonlinear dynamics. Consequently, linear systems cannot represent certain practically important fault types that require nonlinear descriptions.

The purpose of the next two parts consists in the extension of these ideas towards two classes of nonlinear dynamical systems, namely towards Hammerstein-Wiener systems (Part II), as well as towards piecewise affine systems (Part III). The extensions preserve many of the main ideas and advantages of the fault-hiding principle, but due to the absence of the superposition principle in nonlinear systems, certain substantial differences are unavoidable. The process control application studied in [177] is revisited in Part IV of this monograph, where advances and improvements of the new nonlinear methods over the linear methods are demonstrated.

# Part II
# Reconfigurable Control of Hammerstein-Wiener Systems

In this part, the reconfigurable control problem is solved for the class of Hammerstein-Wiener systems. The main motivation for studying this class of systems is the presence of saturation constraints on the input signals. Actuator and sensor faults may occur simultaneously in this approach. Solutions are given with respect to the problems of recovering closed-loop stability and asymptotic tracking. All solutions are based on the fault-hiding principle.

# Chapter 5
# Control Reconfiguration Problem for Hammerstein-Wiener Systems

**Abstract.** This chapter defines Hammerstein-Wiener systems and the nominal closed-loop system. It is shown how faults are modelled in Hammerstein-Wiener systems, and the reconfiguration problem is stated for the class of Hammerstein-Wiener systems. Bibliographic notes on these systems conclude the chapter.

## 5.1 Nominal Hammerstein-Wiener Systems

Hammerstein-Wiener systems are an extension of linear systems that can express static nonlinear aspects of the system.

**Definition 5.1 (Hammerstein-Wiener system [150]).** A *Hammerstein-Wiener system* is a system of first-order ODEs

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{h}(\boldsymbol{C}\boldsymbol{x}(t)) \\ \boldsymbol{z}(t) &= \boldsymbol{h}_z(\boldsymbol{C}_z\boldsymbol{x}(t)), \end{cases} \tag{5.1}$$

where all signals are in accordance with Definition 3.2, all matrices are in accordance with Definition 4.1, and where $\boldsymbol{\varphi} : \mathbb{R}^m \to \mathbb{R}^m$ is a memoryless nonlinear input function, $\boldsymbol{h} : \mathbb{R}^n \to \mathbb{R}^q$ is a memoryless nonlinear function for the measured output, $\boldsymbol{h}_z : \mathbb{R}^n \to \mathbb{R}^p$ (Fig. 5.1) is a memoryless nonlinear function for the controlled output, and

$$\boldsymbol{x}(0) = \boldsymbol{x}_0 \tag{5.2}$$

is the initial condition. ◇

Hammerstein-Wiener systems are adequate system models whenever the nonlinear system under consideration may be approximately described by linear dynamics and separate nonlinear distortions of the input and output channels. This viewpoint may arise from physical insight into the system, such as it is the case when considering

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 89–95.
springerlink.com 

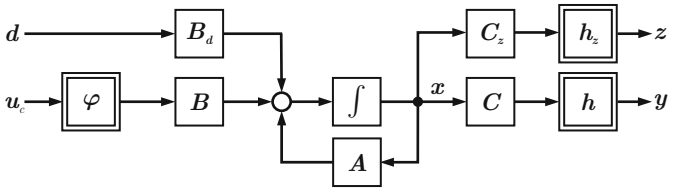


**Fig. 5.1** Hammerstein-Wiener system.

the saturations that typically apply to inputs in technological systems. For the outputs, such a perspective arises, for example, when a measured quantity is a nonlinear function of a state variable, such as the electrical conductivity of salt water, which depends on the salt concentration and on the temperature of the water.

The class of saturated systems is obtained from the class of Hammerstein-Wiener systems by choosing the decomposed saturation function (2.1) as input nonlinearity ($\boldsymbol{\varphi}(\cdot) = \text{sat}(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \cdot)$), and the identity function as output nonlinearities ($\boldsymbol{h}(\boldsymbol{y}) = \boldsymbol{y}$, $\boldsymbol{h}_z(\boldsymbol{z}) = \boldsymbol{z}$).

**Definition 5.2 (Saturated system).** A *saturated system* is a system of first-order ODEs

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\,\text{sat}\left(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u}_c(t)\right) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t), \end{cases} \tag{5.3}$$

where all signals are in accordance with Definition 3.2, all matrices are in accordance with Definition 4.1, where the saturation function $\boldsymbol{u}_s = \text{sat}(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u})$,

$$u_{s,i} \triangleq \begin{cases} \underline{u}_i & \text{if } u_i \le \underline{u}_i \\ u_i & \text{if } \underline{u}_i \le u_i \le \overline{u}_i \\ \overline{u}_i & \text{if } \overline{u}_i \le u_i, \end{cases}$$

was first defined in Equation (2.1), and where $\underline{\boldsymbol{u}}$ and $\overline{\boldsymbol{u}}$ are vectors defining lower and upper input signal range limits. The initial condition is (5.2). ◇

**Example 5.1 (Hammerstein model of a ship).** *Since the ship has input range limits, a Hammerstein model of the ship model (1.1)–(1.4) has the form of a saturated system (5.3). The parameters (4.8), (4.9) of the linearised model are used for the linear dynamical part of the model, and the input range limits are* $\underline{\boldsymbol{u}} = (-1\ -1\ -1)^T$ *and* $\overline{\boldsymbol{u}} = (1\ 1\ 1)^T$.

## 5.2 Nominal Closed-Loop System and Assumptions

Consider the nominal Hammerstein-Wiener controller $\Sigma_C$ for the nominal Hammerstein-Wiener plant (5.1),

$$\Sigma_C : \begin{cases} \dot{\boldsymbol{x}}_c(t) &= \boldsymbol{A}_c\boldsymbol{x}_c(t) + \boldsymbol{B}_c\boldsymbol{\varphi}_y(\boldsymbol{y}(t)) + \boldsymbol{E}_c\boldsymbol{\varphi}_r(\boldsymbol{r}(t)) \\ \hat{\boldsymbol{u}}_c(t) &= \boldsymbol{C}_c\boldsymbol{x}_c(t) + \boldsymbol{D}_c\boldsymbol{\varphi}_y(\boldsymbol{y}(t)) + \boldsymbol{F}_c\boldsymbol{\varphi}_r(\boldsymbol{r}(t)) \\ \boldsymbol{u}_c(t) &= \boldsymbol{h}_u(\hat{\boldsymbol{u}}_c(t)) \end{cases} \quad (5.4)$$
$$\boldsymbol{x}_c(0) = \boldsymbol{x}_{c0}$$

with the internal state $\boldsymbol{x}_c(t) \in \mathbb{R}^{n_c}$, the reference input $\boldsymbol{r}(t) \in \mathbb{R}^q$, the nonlinear input functions $\boldsymbol{\varphi}_y(\cdot)$ and $\boldsymbol{\varphi}_r(\cdot)$, and the nonlinear output function $\boldsymbol{h}_u(\cdot)$. The nominal Hammerstein-Wiener plant (5.1) together with the nominal controller (5.4) gives rise to the nominal closed-loop system $(\Sigma_P, \Sigma_C)$.

The following assumptions about the system (5.1) are in place throughout this chapter.

**Assumption 5.1 (Sector-bounded Lipschitz input function).** *The function $\boldsymbol{\varphi}(\cdot)$ in the system (5.1) is assumed to be globally uniformly Lipschitz, to be decomposed $\left(\boldsymbol{\varphi}(\boldsymbol{u}_c) = (\varphi_1(u_{c1}), \ldots, \varphi_m(u_{cm}))^T\right)$ and to lie within a sector $[0, \boldsymbol{k}_\varphi]$ with the vector of upper sector bounds $\boldsymbol{k}_\varphi = (k_{\varphi 1}, \ldots, k_{\varphi m})^T$.*

**Assumption 5.2 (Sector-bounded Lipschitz output function).** *The function $\boldsymbol{h}(\cdot)$ in the system (5.1) is assumed to be globally uniformly Lipschitz, to be decomposed $\left(\boldsymbol{h}(\boldsymbol{y}) = (h_1(y_1), \ldots, h_q(y_q))^T\right)$, and to lie within a sector $[0, \boldsymbol{k}_h]$ with the vector of upper sector bounds $\boldsymbol{k}_h = (k_{h1}, \ldots, k_{hq})^T$.*

The previous two assumptions exclude functions with infinite gain, as well as functions whose output reverse the sign of the input.

The sector bounds $\boldsymbol{k}_\varphi$ and $\boldsymbol{k}_h$ of the decomposed functions $\boldsymbol{\varphi}(\cdot)$ and $\boldsymbol{h}(\cdot)$ are expressed in the diagonal matrices $\boldsymbol{S}_\varphi$ and $\boldsymbol{S}_h$ as follows:

$$\boldsymbol{S}_\varphi \triangleq \operatorname{diag}(S_{\varphi i}) \text{ where } S_{\varphi i} = 1/k_{\varphi i}, \quad i = 1, \ldots, m \quad (5.5)$$
$$\boldsymbol{S}_h \triangleq \operatorname{diag}(S_{hi}) \text{ where } S_{hi} = 1/k_{hi}, \quad i = 1, \ldots, q. \quad (5.6)$$

**Assumption 5.3 (Stabilising and setpoint tracking nominal control).** *The nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$, formed by the nominal Hammerstein-Wiener plant (5.1) and the nominal Hammerstein-Wiener controller (5.4), is IOS with respect to the input $(\boldsymbol{r}, \boldsymbol{d})$ and the output $(\boldsymbol{x}, \boldsymbol{u}_c)$. Furthermore, the tracking error*

$$\boldsymbol{e}_z(t) \triangleq \boldsymbol{r}(t) - \boldsymbol{z}(t) \quad (5.7)$$

*satisfies certain desirable steady-state tracking and performance properties for arbitrary initial conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_{c0}$.*

The previous assumption means that bounded reference and disturbance inputs lead to bounded state variables and control inputs. This assumption is not restrictive in technological control applications. The achievable nominal tracking properties depend on the nonlinear functions of the plant. For saturated systems in particular, tracking and disturbance rejection are local properties that are achievable only for reference signals and disturbances that lie inside certain regions of attraction [185].

## 5.3 Faults in Hammerstein-Wiener Systems

In this section, faults are introduced into the Hammerstein-Wiener system model (5.1).

**Definition 5.3 (Actuator faults in Hammerstein-Wiener systems).** An *actuator fault* in a Hammerstein-Wiener system is an event occurring at time $t_f$ that changes the nominal input matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ to the faulty input matrix $\boldsymbol{B}_f \in \mathbb{R}^{n\times m}$ of the same dimensions, and the nominal nonlinear input function $\boldsymbol{\varphi}(\cdot)$ to the faulty nonlinear input function $\boldsymbol{\varphi}_f(\cdot)$. ◇

**Definition 5.4 (Sensor faults in Hammerstein-Wiener systems).** A *sensor fault* in a Hammerstein-Wiener system is an event occurring at time $t_f$ that changes the nominal measurement matrix $\boldsymbol{C} \in \mathbb{R}^{q\times n}$ to the faulty measurement matrix $\boldsymbol{C}_f \in \mathbb{R}^{q\times n}$ of the same dimensions, and the nominal nonlinear output function $\boldsymbol{h}(\cdot)$ to the faulty nonlinear output function $\boldsymbol{h}_f(\cdot)$. ◇

Therefore, the nominal Hammerstein-Wiener system (5.1) changes to the faulty Hammerstein-Wiener system

$$\Sigma_{Pf}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{\varphi}_f\left(\boldsymbol{u}_f(t)\right) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{h}_f(\boldsymbol{C}_f\boldsymbol{x}_f(t)) \\ \boldsymbol{z}_f(t) &= \boldsymbol{h}_z(\boldsymbol{C}_z\boldsymbol{x}_f(t)) \end{cases} \qquad \boldsymbol{x}_f(0) = \boldsymbol{x}_0. \tag{5.8}$$

The modification of the input and output matrices $\boldsymbol{B}_f$ and $\boldsymbol{C}_f$ is the same as for linear systems described in Section 4.3, also concerning the blockage off the operating point. The nonlinear input function $\boldsymbol{\varphi}_f(\cdot)$ and the nonlinear output function $\boldsymbol{h}_f(\cdot)$ reflect, for instance, changed nonlinear actuator or sensor characteristics or modified actuation range. Table 5.1 summarises the expressiveness of the Hammerstein-Wiener fault model. Clearly, more fault types are representable by these systems than by linear systems. The fault types expressible in Hammerstein-Wiener systems but not in linear systems are highlighted by means of circled checkmarks.

In particular, the saturated system (5.3) changes to the faulty saturated system

$$\Sigma_{Pf}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\,\mathrm{sat}\left(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f(t)\right) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t), \end{cases} \tag{5.9}$$

where the saturation function is nominal but its range arguments change.

**Example 5.2 (Hammerstein-Wiener model of ship subject to faults).** *The saturated system model of the faulty plant has the same model elements $\boldsymbol{B}_f$, $\boldsymbol{C}_f$, and $\bar{\boldsymbol{d}}_{fp}$ as the linear model (4.19) of the faulty ship. Furthermore, the reduced actuation range of the left thruster ($f_4$) can be modelled without leaving the class of Hammerstein-Wiener systems. The range of the left thruster is reduced to* ±60%,

**Table 5.1** Technological faults representable by Hammerstein-Wiener fault models.

| Technological fault | Representable | By model parameter |
|---|---|---|
| Changed actuator gain | ✓ | $\boldsymbol{B}_f$ |
| Changed nonlinear actuator characteristic | ⓥ | $\boldsymbol{\varphi}_f(\cdot)$ |
| Changed or reduced actuation range | ⓥ | $\boldsymbol{\varphi}_f(\cdot)$ |
| Actuator failure at the operating point | ✓ | $\boldsymbol{B}_f$ |
| Actuator failure off the operating point | (✓) | $\sum_{j\in\mathcal{A}} \boldsymbol{b}_j \bar{u}_j$ (affine) |
| Changed sensor gain | ✓ | $\boldsymbol{C}_f$ |
| Changed nonlinear sensor characteristic | ⓥ | $\boldsymbol{h}_f(\cdot)$ |
| Sensor failure | ✓ | $\boldsymbol{C}_f$ |

Legend: ✓: fully representable; ⓥ: exclusively representable in this system class; (✓): representable leaving the system class; ×: not representable.

$u_1 \in [-0.6\ 0.6]$. *This reduction is modelled by means of a modified saturation function* $\mathrm{sat}(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f)$, *where* $\underline{\boldsymbol{u}}_f = \begin{pmatrix}-0.6 & -1 & -1\end{pmatrix}$ *and* $\overline{\boldsymbol{u}}_f = \begin{pmatrix}0.6 & 1 & 1\end{pmatrix}$. *The entire saturated faulty system is described by the equations*

$$\begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\,\mathrm{sat}(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f(t)) + \bar{\boldsymbol{d}}_f p + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ z_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t). \end{cases} \tag{5.10}$$

*The fault* $f_3$ *meaning a floating rudder is obtained by the special case* $p = 0$.

## 5.4 Specific Reconfiguration Problems

The reconfiguration block (3.21) is a nonlinear dynamical system consisting of a Hammerstein-Wiener virtual actuator and a Hammerstein-Wiener virtual sensor. The virtual sensor is strictly speaking not a Hammerstein-Wiener system because its dynamics include output error injection. The reconfiguration block (3.21) has the form

$$\Sigma_R : \begin{cases} \dot{\boldsymbol{\zeta}}(t) &= \boldsymbol{A}_r\boldsymbol{\zeta}(t) + \boldsymbol{\varphi}_{r\zeta}(\boldsymbol{\zeta}(t)) + \boldsymbol{B}_r\boldsymbol{\varphi}_{ru}(\boldsymbol{u}_c(t)) + \boldsymbol{E}_r\boldsymbol{y}_f(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{h}_{r\zeta}(\boldsymbol{C}_r\boldsymbol{\zeta}(t)) + \boldsymbol{h}_{ry}\left(\boldsymbol{F}_r\boldsymbol{y}_f(t)\right) \\ \boldsymbol{u}_f(t) &= \boldsymbol{G}_r\boldsymbol{\zeta}(t) + \boldsymbol{H}_r\boldsymbol{u}_c(t), \\ \boldsymbol{\zeta}(0) &= \boldsymbol{\zeta}_0 \end{cases} \tag{5.11}$$

that is connected to the faulty Hammerstein-Wiener plant (5.8) through their common variables $\boldsymbol{u}_f$ and $\boldsymbol{y}_f$ as well as to the nominal controller (5.4) through the common variable $\boldsymbol{u}_c$ and the interconnection $\boldsymbol{y}(t) = \boldsymbol{y}_c(t)$. The faulty plant, the reconfiguration block, and the nominal controller form the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_R, \Sigma_C)$.

The reconfiguration problems 3.3, 3.4, and 3.5 stated in Chapter 3 are now specified for Hammerstein-Wiener systems, starting with stability.

**Problem 5.1 (Stability recovery for Hammerstein-Wiener systems).** Consider the nominal Hammerstein-Wiener plant $\Sigma_P$ defined in Equation (5.1) and the faulty Hammerstein-Wiener plant $\Sigma_{Pf}$ defined in Equation (5.8). Find a reconfigured block $\Sigma_R$ of the form (5.11) such that

$$\forall \Sigma_C : \{(\Sigma_P, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\}.$$

Problem 3.4 will be considered for the special class of saturated systems, which form a subclass of Hammerstein systems.

**Problem 5.2 (Stable asymptotic setpoint tracking recovery for saturated systems).** Consider the nominal saturated plant $\Sigma_P$ defined in Equation (5.3) and the faulty saturated plant $\Sigma_{Pf}$ defined in Equation (5.9). Find a reconfiguration block $\Sigma_R$ of the form (5.11) such that for all nominal controllers $\Sigma_C$ satisfying Assumption 4.1, the reconfigured closed-loop system is ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$, namely

$$\forall \Sigma_C : \{(\Sigma_P, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\},$$

and such that for constant disturbances $\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t)$, constant reference inputs $\boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$, and arbitrary initial conditions $\boldsymbol{x}_0$ and $\boldsymbol{x}_{c0}$ of the plant and the controller, it holds that

$$\left\{ \lim_{t\to\infty} (\boldsymbol{r}(t) - \boldsymbol{z}(t)) = \boldsymbol{0} \right\} \Rightarrow \left\{ \lim_{t\to\infty} \left( \boldsymbol{r}(t) - \boldsymbol{z}_f(t) \right) = \boldsymbol{0} \right\}. \tag{5.12}$$

**Problem 5.3 (Optimal stable performance recovery for saturated systems).** Consider the nominal saturated plant $\Sigma_P$ defined in Equation (5.3) and the faulty saturated plant $\Sigma_{Pf}$ with initial condition $\boldsymbol{x}_0$ defined in Equation (5.9). Find an optimal reconfiguration block $\Sigma_{R*}$ such that the reconfigured closed-loop system is ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$, and such that, for all nominal controllers $\Sigma_C$ satisfying Assumption 5.3, a tradeoff $\lambda\gamma_z + (1-\lambda)\gamma_u$ parameterized over $\lambda \in [0,1]$ is attained between

1. Optimal approximation of the closed-loop output trajectory in the sense that if $\boldsymbol{z}_f^*$ is the output obtained with $\Sigma_{R*}$ and $\boldsymbol{z}_f$ is the output obtained with any other reconfiguration block $\Sigma_R$, then

$$\forall \boldsymbol{u}_c(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \gamma_z = \frac{\|\boldsymbol{z}(t) - \boldsymbol{z}_f^*(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \leq \frac{\|\boldsymbol{z}(t) - \boldsymbol{z}_f(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \tag{5.13}$$

   holds, and
2. Minimal amplification of the input signal in the sense that if $\boldsymbol{u}_f^*$ is the control input obtained with $\Sigma_{R*}$ and $\boldsymbol{u}_f$ is the control input obtained with any other reconfiguration block $\Sigma_R$, then

$$\forall \boldsymbol{u}_c(t), \boldsymbol{x}_0, \boldsymbol{x}_{c0}, \boldsymbol{\zeta}_0 : \ 1 \leq \gamma_u = \frac{\|\boldsymbol{u}_f^*(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}} \leq \frac{\|\boldsymbol{u}_f(t)\|_{\mathcal{L}_2}}{\|\boldsymbol{u}_c(t)\|_{\mathcal{L}_2}}. \tag{5.14}$$

$\diamond$

The solution to Problems 5.1, 5.2, and 5.3, will be discussed in Chapters 6, 7, and 8, respectively. The stability recovery for Hammerstein systems based on fault-hiding principles was first reported in [172]. The tracking recovery methods are first reported in [174].

## 5.5 Bibliographic Notes on Hammerstein-Wiener Systems

Hammerstein-Wiener systems are linear dynamical systems with static nonlinear functions acting on the input and output signals. They have been used, for example, to model the magnetosphere around the earth [150], to model pH neutralisation processes [154], distillation columns, heat exchangers, stirred reactors, and other nonlinear industrial processes [22].

Several approaches to the identification of Hammerstein-Wiener systems have been reported in the literature. A three-stage iterative least-squares approach that first estimates the Hammerstein-subsystem, second identifies the Wiener-nonlinearity, and third optimises the selection of nonlinear basis functions from a candidate set is described in [150]. The method also accounts for missing data. Other approaches are based on subspace identification and least-squares support vector machines [69]. Further related literature may be found within the two previous references.

To control Hammerstein-Wiener systems, several approaches have been taken that typically depend on further assumptions about the nonlinear functions. If these functions are invertible, linearising control has been applied that consists of simple inverse functions of $\boldsymbol{f}$ and $\boldsymbol{h}$ placed in the input and output channels. The remaining linear subsystem may be controlled by means of any suitably designed linear controller [150]. Alternatively, model-predictive control has been applied, where the main difficulty consists in formulating the control problem in such a way that the underlying optimisation problem is convex [22].

The presence of saturations in the Hammerstein nonlinearity complicates the control problem [185]. The stability analysis in terms of necessary and sufficient conditions and the synthesis of stabilising controllers for saturated systems are known to be undecidable problems for saturated systems, unless the system matrix is symmetric [23].

This review shows that Hammerstein-Wiener systems are practically relevant, and that their control is nontrivial. The reconfigurable control of these systems has not been studied before to the author's knowledge. The undecidability property implies that necessary and sufficient reconfigurability conditions are not to be expected, and are indeed not achieved in this monograph. However, sufficient stabilisability conditions are provided in this monograph along with synthesis algorithms for recovering the nominal closed-loop stability, tracking, and performance properties.

# Chapter 6
# Stability Recovery after Actuator and Sensor Faults in Hammerstein-Wiener Systems

**Abstract.** This chapter provides the solution to the stability recovery problem after combined actuator and sensor faults in Hammerstein-Wiener systems. The structure of the reconfiguration block is defined, and sufficient conditions are stated that guarantee the input-to-state stability of the reconfigured closed-loop system. These conditions are also used for the synthesis of the free reconfiguration block parameters. The special cases of pure actuator or sensor faults that give rise to simplified reconfiguration blocks are separately discussed. These solutions are shown to be robust against uncertain models of the faulty system, and they are shown to be universal solutions to the stated reconfiguration problems.

## 6.1 Hammerstein-Wiener Virtual Actuator and Virtual Sensor

For the combined occurrence of actuator and sensor faults ($\boldsymbol{B}_f \neq \boldsymbol{B}$, $\boldsymbol{C}_f \neq \boldsymbol{C}$), the reconfiguration block (5.11) is realised by the interconnection of the following virtual sensor $\Sigma_S$ and virtual actuator $\Sigma_A$ (Fig. 6.1).

**Definition 6.1 (Hammerstein-Wiener virtual sensor).** The *Hammerstein-Wiener virtual sensor* is the dynamical system

$$\Sigma_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= \boldsymbol{A}\hat{\boldsymbol{x}}_f(t) + \boldsymbol{B}_f\boldsymbol{\varphi}_f\left(\boldsymbol{u}_f(t)\right) + \boldsymbol{L}(\boldsymbol{y}_f(t) - \hat{\boldsymbol{y}}_f(t)) \\ \hat{\boldsymbol{y}}_f(t) &= \boldsymbol{h}_f\left(\boldsymbol{C}_f\hat{\boldsymbol{x}}_f(t)\right) \end{cases} \tag{6.1}$$

with the initial condition $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$. ◇

As in the linear case, the Hammerstein-Wiener virtual sensor is essentially an observer for the state of the faulty plant. It consists of a model of the faulty plant augmented by output error injection.

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 97–118.
springerlink.com

**Definition 6.2 (Hammerstein-Wiener virtual actuator).** The *Hammerstein-Wiener virtual actuator* is the dynamical system

$$\Sigma_A : \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) & = \boldsymbol{A}\tilde{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) \\ \boldsymbol{u}_f(t) & = \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) & = \boldsymbol{h}(\boldsymbol{C}\tilde{\boldsymbol{x}}(t)) \end{cases} \tag{6.2}$$

with the initial condition $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_{f0}$, and where $\boldsymbol{x}_{\Delta}(t) = \tilde{\boldsymbol{x}}(t) - \hat{\boldsymbol{x}}_f(t)$ in accordance with Equation (3.35). ◇

The virtual actuator contains a reference model for the nominal dynamics along with feedback of the difference between the reference state and the observed state, as well as feedthrough of the control input.

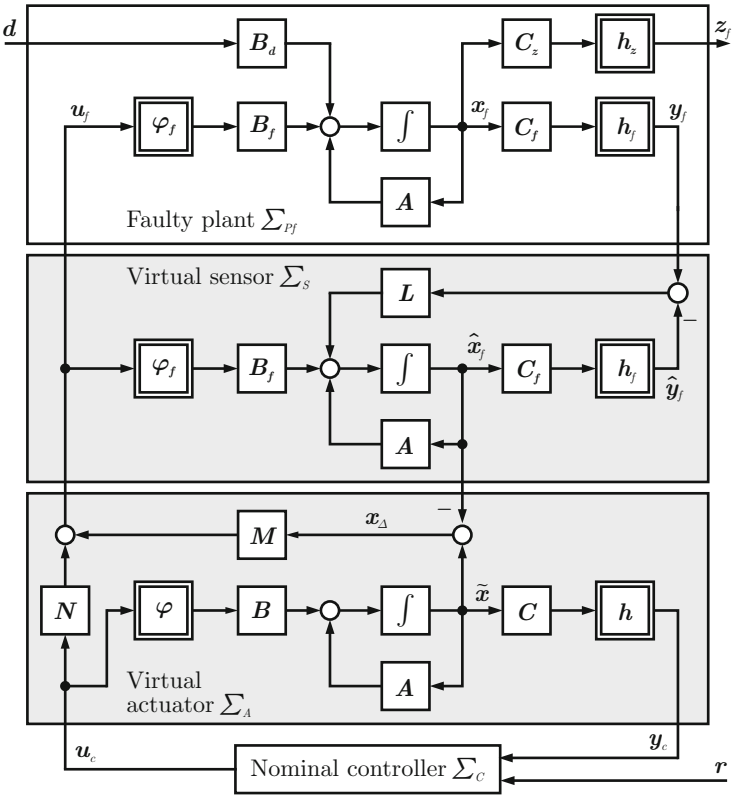


**Fig. 6.1** Reconfiguration block $\Sigma_R = (\Sigma_S, \Sigma_A)$ for reconfigurable control after the combined occurrence of actuator and sensor faults in Hammerstein-Wiener systems.

In order to satisfy the inactivity conditions (3.22), (3.23) imposed on reconfiguration blocks prior to reconfiguration time, the parameters of the Hammerstein-Wiener virtual actuator have to satisfy $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{N} = \boldsymbol{I}$, $\boldsymbol{y}_c(t) = \boldsymbol{y}_f(t)$ before reconfiguration. At reconfiguration time, the parameters $\boldsymbol{M}$ and $\boldsymbol{N}$ are determined based on the model of the faulty plant, and the controller is connected to the output $\boldsymbol{y}_c(t) = \boldsymbol{C}\tilde{\boldsymbol{x}}(t)$ of the virtual actuator.

As a preliminary result, the reconfiguration block consisting of the Hammerstein-Wiener virtual sensor and virtual actuator satisfies the weak fault-hiding goal, as the following Lemma shows.

**Lemma 6.1 (Weak fault-hiding).** *The reconfigured plant $(\Sigma_{Pf}, \Sigma_S, \Sigma_A)$ that consists of the faulty Hammerstein-Wiener plant (5.8), the Hammerstein-Wiener virtual sensor (6.1), and the Hammerstein-Wiener virtual actuator (6.2) satisfies the weak fault hiding goal.*

*Proof.* After transformation into the coordinates of the observation error $\boldsymbol{e}$ and difference state $\boldsymbol{x}_\Delta$, the model of the reconfigured plant is given by the equations

$$\begin{aligned}\begin{pmatrix}\dot{\tilde{\boldsymbol{x}}}(t)\\ \dot{\boldsymbol{e}}(t)\\ \dot{\boldsymbol{x}}_\Delta(t)\end{pmatrix} &= \begin{pmatrix}\boldsymbol{A}\tilde{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t))\\ \boldsymbol{A}\boldsymbol{e}(t) - \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)\\ \boldsymbol{A}\boldsymbol{x}_\Delta(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) - \boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{N}\boldsymbol{u}_c(t))\end{pmatrix}\\ &\quad - \begin{pmatrix}\boldsymbol{0}\\ \boldsymbol{L}\\ \boldsymbol{L}\end{pmatrix}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t) - \boldsymbol{e}(t))) - \boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t)))\Big) \qquad (6.3)\end{aligned}$$

$$\boldsymbol{y}_c(t) = \boldsymbol{h}\left(\begin{pmatrix}\boldsymbol{C} & \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}\tilde{\boldsymbol{x}}(t)\\ \boldsymbol{e}(t)\\ \boldsymbol{x}_\Delta(t)\end{pmatrix}\right), \quad \begin{pmatrix}\tilde{\boldsymbol{x}}(0)\\ \boldsymbol{e}(0)\\ \boldsymbol{x}_\Delta(0)\end{pmatrix} = \begin{pmatrix}\hat{\boldsymbol{x}}_{f0}\\ \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0\\ \boldsymbol{0}\end{pmatrix}. \qquad (6.4)$$

This model shows that the dynamical equation for the reference state $\tilde{\boldsymbol{x}}$ is decoupled from the observation error $\boldsymbol{e}$ and the difference state $\boldsymbol{x}_\Delta$. The output $\boldsymbol{y}_c$ depends only on the state $\tilde{\boldsymbol{x}}$. Stronger fault-hiding goals are not achievable, since the disturbance behaviour is not recovered and the correct initial condition is generally unknown ($\hat{\boldsymbol{x}}_{f0} \neq \boldsymbol{x}_0$), due to the general lack of measurements of the entire system state. ■

**Example 6.1 (Hammerstein-Wiener virtual actuator and virtual sensor for the ship).** *The linear parts of the Hammerstein-Wiener virtual actuator (6.2) and the Hammerstein-Wiener virtual sensor (6.1) are defined through the model elements (4.8), (4.9). The nonlinear input function $\boldsymbol{\varphi}$ is a saturation function (2.1) with the vectors of lower bounds $\underline{\boldsymbol{u}} = (-1\ -1\ -1)^T$ and upper bounds $\overline{\boldsymbol{u}} = (1\ 1\ 1)^T$ as defined in Example 5.1. The nonlinear output function $\boldsymbol{h}$ is the identity, the virtual actuator is a Hammerstein virtual actuator, and the virtual sensor is a Hammerstein virtual sensor.*

## 6.2 Main Stability Result

The main result consists in a theorem that provides a sufficient condition for the solvability of Problem 5.1 and a design procedure for finding stabilising parameters $\boldsymbol{L}$ and $\boldsymbol{M}$ for the Hammerstein-Wiener virtual sensor and virtual actuator. The main theorem regarding reconfigured closed-loop stability uses the following partial results that separately concern the observation error system

$$\Sigma_e : \begin{cases} \dot{\boldsymbol{e}}(t) = & \boldsymbol{A}\boldsymbol{e}(t) - \boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t))) - \boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)))\Big) \\ & -\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ \boldsymbol{e}(0) = \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0 \end{cases} \tag{6.5}$$

and the difference system

$$\Sigma_{\Delta} : \begin{cases} \dot{\boldsymbol{x}}_{\Delta}(t) = & \boldsymbol{l}(\boldsymbol{x}_{\Delta}(t), \boldsymbol{u}_c(t)) \\ & -\boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t))) - \boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)))\Big) \end{cases} \tag{6.6}$$

$$\boldsymbol{l}(\boldsymbol{x}_{\Delta}, \boldsymbol{u}_c) \triangleq \boldsymbol{A}\boldsymbol{x}_{\Delta} + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c) - \boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{M}\boldsymbol{x}_{\Delta} + \boldsymbol{N}\boldsymbol{u}_c) \tag{6.7}$$

$$\boldsymbol{x}_{\Delta}(0) = \boldsymbol{0}.$$

**Lemma 6.2 (Input-to-state stability of the observation error).** *Consider the dynamics of the observation error (6.5), where the diagonal matrix $\boldsymbol{S}_h \in \mathbb{R}^{q\times q}$, $\boldsymbol{S}_h \geq 0$ reflects the sector bound (5.6) of $\boldsymbol{h}_f(\cdot)$. Suppose that Assumption 5.3 holds and that $\boldsymbol{A}$ is Hurwitz. The observation error system $\Sigma_e$ is globally uniformly exponentially stable for $\boldsymbol{d} \equiv \boldsymbol{0}$ and ISS w.r.t. its input $\boldsymbol{d}$, if there exists a symmetric positive definite matrix $\boldsymbol{X}_s = \boldsymbol{X}_s^T > 0$, $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$, and a matrix $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$ such that the LMI*

$$\left(\begin{array}{c|c} -(\boldsymbol{A}^T\boldsymbol{X}_s + \boldsymbol{X}_s\boldsymbol{A}) & \boldsymbol{C}_f^T - \boldsymbol{Y}_s \\ \hline \star & 2\boldsymbol{S}_h \end{array}\right) > 0 \tag{6.8}$$

*is satisfied, where $\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$.*

*Proof.* The observation error system (6.5) is shown in Fig. 6.2. The Lipschitz condition on $\boldsymbol{h}_f(\cdot)$ stated in Assumption 5.2 guarantees that

$$\forall \boldsymbol{x}_f, \boldsymbol{e} : \ \|\boldsymbol{h}_f(\boldsymbol{x}_f) - \boldsymbol{h}_f(\boldsymbol{x}_f + \boldsymbol{e})\| \leq R\|\boldsymbol{e}\|. \tag{6.9}$$

The remaining stability proof consists of the verification of exponential stability of the unforced observation error, and the subsequent verification of the ISS property for the forced observation error. From Theorem 2.5, it follows directly that the observation error is asymptotically stable for $\boldsymbol{d} \equiv \boldsymbol{0}$. In fact, the unforced observation error is globally uniformly exponentially stable [97, Proof of Theorem 7.1].

From the exponential stability of the unforced observation error and from Theorem 2.1, it follows that the unforced observation error has a Lyapunov

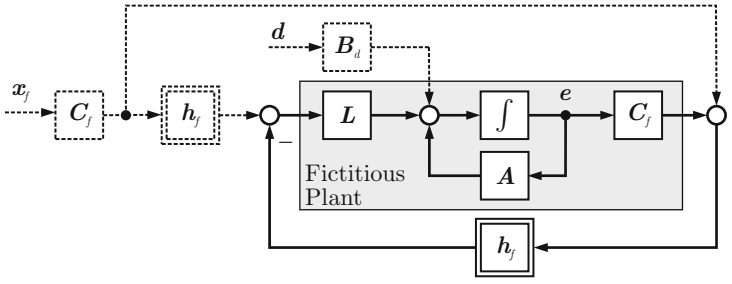


**Fig. 6.2** Nonlinear observation error system.

function $V(\boldsymbol{e})$. The derivative of $V(\boldsymbol{e})$ with respect to the forced observation error system (6.5) satisfies the relations

$$\begin{aligned}\dot{V}(\boldsymbol{e}) &= \nabla V(\boldsymbol{e})\Big(\boldsymbol{A}\boldsymbol{e}(t) + \boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f\boldsymbol{x}_f(t)) - \boldsymbol{h}_f(\boldsymbol{C}_f(\boldsymbol{x}_f(t)+\boldsymbol{e}(t)))\Big) - \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)\Big) \\ &\leq -c_3\|\boldsymbol{e}\|^2 + c_4R\|\boldsymbol{e}\|^2\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\| + c_4\|\boldsymbol{e}\|\cdot\|\boldsymbol{B}_{\boldsymbol{d}}\|\cdot\|\boldsymbol{d}\|\end{aligned} \tag{6.10}$$

for $c_3, c_4 > 0$ due to the Lipschitz property (6.9) and Theorem 2.1. The first term $-c_3\|\boldsymbol{e}\|^2$ is obviously always negative. Using the parameter $\theta \in (0,1)$, the inequality is rewritten as

$$\begin{aligned}\dot{V}(\boldsymbol{e}) &\leq -c_3(1-\theta)\|\boldsymbol{e}\|^2 - c_3\theta\|\boldsymbol{e}\|^2 + c_4R\|\boldsymbol{e}\|^2\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\| + c_4\|\boldsymbol{e}\|\cdot\|\boldsymbol{B}_{\boldsymbol{d}}\|\cdot\|\boldsymbol{d}\| \\ &\leq -(1-\theta)c_3\|\boldsymbol{e}\|^2, \quad \text{if } \|\boldsymbol{e}\| \geq \frac{c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|}{\theta c_3 - c_4R\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|}\|\boldsymbol{d}\|.\end{aligned}$$

Theorem 2.2 is used with $\alpha_1(r) = c_1r^2$, $\alpha_2(r) = c_2r^2$, $W(\boldsymbol{e}) = (1-\theta)c_3\|\boldsymbol{e}\|^2$ and $\rho(r) = (c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|/(c_3\theta - c_4R\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|))r$ to conclude that the observation error system (6.5) is globally ISS w.r.t. the input $\boldsymbol{d}$, where $\gamma(r) = \sqrt{c_2/c_1}(c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|/(c_3\theta - c_4R\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|))r$ for $c_3\theta - c_4R\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\| > 0$. This completes the proof. ■

The following lemma concerns the difference system.

**Lemma 6.3 (Input-to-state stability of the difference system).** *Consider the dynamics of the difference system (6.6), where the diagonal matrix $\boldsymbol{S}_\varphi \in \mathbb{R}^{m\times m}$, $\boldsymbol{S}_\varphi \geq 0$ reflects the sector bound (5.5) of $\boldsymbol{\varphi}_f(\cdot)$. Suppose that Assumption 5.3 holds and that $\boldsymbol{A}$ is Hurwitz. The difference system $\Sigma_\Delta$ is globally uniformly exponentially stable for $\boldsymbol{u}_c \equiv \boldsymbol{0}$ and $\boldsymbol{e} \equiv \boldsymbol{0}$ as well as ISS w.r.t. its input $(\boldsymbol{u}_c, \boldsymbol{e})$, if there exists a symmetric positive definite matrix $\boldsymbol{X}_a = \boldsymbol{X}_a^T > 0$, $\boldsymbol{X}_a \in \mathbb{R}^{n\times n}$, and a matrix $\boldsymbol{Y}_a \in \mathbb{R}^{m\times n}$ such that the LMI*

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T) & \boldsymbol{Y}_a^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{S}_\varphi \end{array}\right) > 0 \tag{6.11}$$

*is satisfied, where $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$.*

*Proof.* The unforced difference system $\Sigma_\Delta$ defined in Equation (6.6) ($\boldsymbol{e} \equiv \boldsymbol{0}$ and $\boldsymbol{u}_c \equiv \boldsymbol{0}$) is a standard linear system with nonlinear feedback that can be graphically represented as shown in Fig. 6.3. Theorem 2.5 implies that the LMI

$$\left(\begin{array}{c|c} -(\boldsymbol{A}^T\boldsymbol{X}+\boldsymbol{X}\boldsymbol{A}) & \boldsymbol{M}^T-\boldsymbol{X}\boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{S}_\varphi \end{array}\right) \succ 0, \quad \boldsymbol{X}=\boldsymbol{X}^T \succ 0 \tag{6.12}$$

is sufficient for the unforced difference system to be globally asymptotically stable. From this LMI, the LMI (6.11) is obtained as follows. Application of Schur complements to the above LMI leads to the equivalent set of LMIs

$$2\boldsymbol{S}_\varphi \succ 0 \text{ and } -(\boldsymbol{A}^T\boldsymbol{X}+\boldsymbol{X}\boldsymbol{A})-(\boldsymbol{M}^T-\boldsymbol{X}\boldsymbol{B}_f)(2\boldsymbol{S}_\varphi)^{-1}(\boldsymbol{M}-\boldsymbol{B}_f^T\boldsymbol{X}) \succ 0.$$

Application of the congruence transformation $\boldsymbol{X}_a \triangleq \boldsymbol{X}^{-1}$ from left and right brings the set of inequalities into equivalent form

$$2\boldsymbol{S}_\varphi \succ 0 \text{ and } -(\boldsymbol{X}_a\boldsymbol{A}^T+\boldsymbol{A}\boldsymbol{X}_a)-(\boldsymbol{X}_a\boldsymbol{M}^T-\boldsymbol{B}_f)(2\boldsymbol{S}_\varphi)^{-1}(\boldsymbol{M}\boldsymbol{X}_a-\boldsymbol{B}_f^T) \succ 0,$$

which is brought into the stated form by means of the substitution $\boldsymbol{Y}_a \triangleq \boldsymbol{M}\boldsymbol{X}_a$ and reverse Schur complements:

$$\left(\begin{array}{c|c} -(\boldsymbol{X}_a\boldsymbol{A}^T+\boldsymbol{A}\boldsymbol{X}_a) & \boldsymbol{Y}_a^T-\boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{S}_\varphi \end{array}\right) \succ 0, \quad \boldsymbol{X}_a=\boldsymbol{X}_a^T \succ 0.$$

From Theorem 2.5, the unforced difference system is globally asymptotically stable and globally uniformly exponentially stable [97, Proof of Theorem 7.1].

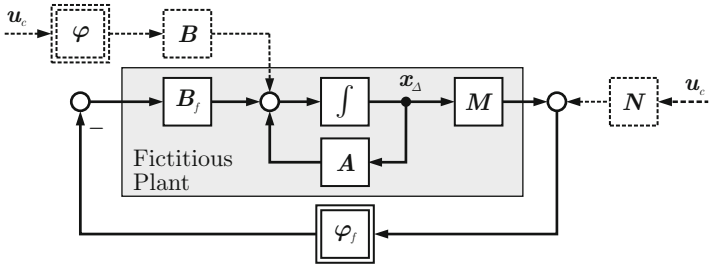


**Fig. 6.3** Nonlinear difference system.

From Theorem 2.1, it follows that the unforced ($\boldsymbol{u}_c \equiv \boldsymbol{0}, \boldsymbol{e} \equiv \boldsymbol{0}$) difference system has a Lyapunov function $V(\boldsymbol{x}_\Delta)$. The derivative of $V(\boldsymbol{x}_\Delta)$ with respect to the forced difference system (6.6) satisfies the relations

$$\begin{aligned} \dot{V}(\boldsymbol{x}_\Delta) =& \nabla V(\boldsymbol{x}_\Delta)\boldsymbol{l}(\boldsymbol{x}_\Delta,\boldsymbol{0}) + \nabla V(\boldsymbol{x}_\Delta)\left(\boldsymbol{l}(\boldsymbol{x}_\Delta,\boldsymbol{u}_c)-\boldsymbol{l}(\boldsymbol{x}_\Delta,\boldsymbol{0})\right) \\ & - \nabla V(\boldsymbol{x}_\Delta)\boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}-\boldsymbol{x}_\Delta-\boldsymbol{e}))-\boldsymbol{h}_f(\boldsymbol{C}_f(\tilde{\boldsymbol{x}}-\boldsymbol{x}_\Delta))\Big) \\ \le & -c_3\|\boldsymbol{x}_\Delta\|^2 + c_4\|\boldsymbol{x}_\Delta\|\|L\|\|\boldsymbol{u}_c\| + c_4R\|\boldsymbol{L}\|\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{x}_\Delta\|\cdot\|\boldsymbol{e}\| \end{aligned} \tag{6.13}$$

for $c_3, c_4 > 0$ due to the Lipschitz property (5.1) and Theorem 2.1, where the function $\boldsymbol{l}$ is defined in Equation (6.7). Using the parameter $\theta \in (0,1)$, the inequality is rewritten as

$$\begin{aligned}\dot{V}(\boldsymbol{x}_{\Delta}) &\leq -(1-\theta)c_3\|\boldsymbol{x}_{\Delta}\|^2 - \theta c_3\|\boldsymbol{x}_{\Delta}\|^2 + c_4\|\boldsymbol{x}_{\Delta}\|\|L\|\boldsymbol{u}_c\| + c_4 R\|\boldsymbol{L}\|\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{x}_{\Delta}\|\cdot\|\boldsymbol{e}\| \\ &\leq -(1-\theta)c_3\|\boldsymbol{x}_{\Delta}\|^2, \quad \text{if } \|\boldsymbol{x}_{\Delta}\| \geq \frac{c_4(L\|\boldsymbol{u}_c\| + R\|\boldsymbol{L}\|\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{e}\|}{c_3\theta}.\end{aligned}$$

Theorem 2.2 is used with $\alpha_1(r) = c_1 r^2$, $\alpha_2(r) = c_2 r^2$, $W(\boldsymbol{x}_{\Delta}) = (1-\theta)c_3\|\boldsymbol{x}_{\Delta}\|^2$, $\rho_u(r) = (c_4 L/(c_3\theta))r$, and $\rho_e(r) = (c_4 R\|\boldsymbol{L}\|\cdot\|\boldsymbol{C}_f\|/(c_3\theta))r$ to conclude that the difference system (6.6) is globally ISS w.r.t. the input $\boldsymbol{u}_c$, where $\gamma_u(r) = \sqrt{c_2/c_1}(c_4 L/c_3\theta)r$, and w.r.t. the input $\boldsymbol{e}$, where $\gamma_e(r) = \sqrt{c_2/c_1}(c_4 R\|\boldsymbol{L}\|\cdot\|\boldsymbol{C}_f\|/c_3\theta)r$. This concludes the proof. ■

The LMI conditions (6.8) and (6.11) each impose a passivity constraint on a certain linear subsystem. The conditions guarantee that the nonlinear input and output functions $\boldsymbol{\varphi}(\cdot)$ and $\boldsymbol{h}(\cdot)$ do not destroy stability. The main result is expressed in the following theorem, which provides the solution to Problem 5.1. Its proof, which uses the Lemmas 6.2 and 6.3, is found in the appendix due to its technical nature.

**Theorem 6.1 (Reconfigured closed-loop stability recovery).** *Consider the reconfigured closed-loop system (5.8), (5.4), (6.1) (6.2) (Fig. 6.1), where the diagonal matrices $\boldsymbol{S}_{\varphi} \in \mathbb{R}^{m\times m}$, $\boldsymbol{S}_{\varphi} \geq 0$, and $\boldsymbol{S}_h \in \mathbb{R}^{q\times q}$, $\boldsymbol{S}_h \geq 0$ reflect the sector bounds (5.5) and (5.6) of $\boldsymbol{\varphi}_f(\cdot)$ and $\boldsymbol{h}_f(\cdot)$. Suppose that Assumption 5.3 holds and that $\boldsymbol{A}$ is Hurwitz. The reconfigured closed-loop system is ISS w.r.t. its input $(\boldsymbol{r}, \boldsymbol{d})$, if there exist symmetric positive definite matrices $\boldsymbol{X}_s = \boldsymbol{X}_s^T > 0$, $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$ and $\boldsymbol{X}_a = \boldsymbol{X}_a^T > 0$, $\boldsymbol{X}_a \in \mathbb{R}^{n\times n}$, as well as matrices $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$ and $\boldsymbol{Y}_a \in \mathbb{R}^{m\times n}$ such that the LMIs (6.8) and (6.11) are satisfied, where $\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$ and $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$.*

*Proof.* See Appendix D, page 259.

*Remark 6.1 (High-gain virtual sensor).* The proof of Lemma 6.2 suggests that a high-gain virtual sensor can reduce the persistent observation error caused by disturbance excitation. The construction of an ISS Lyapunov function for the observation error shows that the observation error $\boldsymbol{e}$ decreases over time whenever

$$\|\boldsymbol{e}\| \geq \frac{c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|}{\theta c_3 - c_4 R\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|}\|\boldsymbol{d}\|.$$

The attraction radius $(c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|)/(\theta c_3 - c_4 R\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|)\|\boldsymbol{d}\|$ for given $\|\boldsymbol{d}\|$ depends on the term $\theta c_3 - c_4 R\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|$, which can have either positive or negative sign. The constants $c_3 > 0$, $c_4 > 0$, $R > 0$, and $\|\boldsymbol{C}_f\|$ are defined through system properties, whereas the gain $\boldsymbol{L}$ is a free parameter that can be used to change the sign of the term $\theta c_3 - c_4 R\cdot\|\boldsymbol{C}_f\|\cdot\|\boldsymbol{L}\|$ from positive to negative. A negative sign means that

the observation error will always decrease. It should, however, be kept in mind that high-gain observers suffer from noise amplification, limiting their practical applicability. ◦

Algorithm 6.1 summarises the procedure of obtaining the reconfiguration solution in a real-time setup. The steps 1-3 describe the nominal closed-loop operation before any faults occur. Once faults are detected in step 4, the design of the Hammerstein-Wiener virtual sensor and the Hammerstein-Wiener virtual actuator occurs in steps 5-7, where the gains $\boldsymbol{L}$ and $\boldsymbol{M}$ are designed. This step always succeeds. After completed gain calculations, the reconfigured closed-loop system is executed in step 8, starting at reconfiguration time $t = 0$.

**Algorithm 6.1.** Stabilising Hammerstein-Wiener virtual actuator and sensor

**Require:** $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{\varphi}, \boldsymbol{h}, \boldsymbol{x}_0, \boldsymbol{x}_{f0}$

1: Initialise the nominal closed-loop system (5.1), (5.4), (6.1) with $\boldsymbol{B}_f = \boldsymbol{B}, \boldsymbol{C}_f = \boldsymbol{C}$, $\boldsymbol{\varphi}_f = \boldsymbol{\varphi}$, $\boldsymbol{h}_f = \boldsymbol{h}$, $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{L} = \boldsymbol{0}$, $\boldsymbol{N} = \boldsymbol{I}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, $\hat{\boldsymbol{x}}_f(t_0) = \boldsymbol{x}_{f0}$, $\tilde{\boldsymbol{x}}(0) = \boldsymbol{x}_{f0}$, deactivate the reconfiguration block by setting $\boldsymbol{y}_c(t) = \boldsymbol{y}(t)$
2: **repeat**
3: Run the nominal closed-loop system
4: **until** actuator or sensor fault $f$ detected and isolated
5: Construct the actuator fault model $\boldsymbol{B}_f$, $\boldsymbol{\varphi}_f$, the sensor fault model $\boldsymbol{C}_f$, $\boldsymbol{h}_f$, the sector bounds $\boldsymbol{S}_{\boldsymbol{\varphi}}$, $\boldsymbol{S}_h$, and update the Hammerstein-Wiener virtual sensor (6.1) and the Hammerstein-Wiener virtual actuator (6.2)
6: Compute feasible solutions $(\boldsymbol{X}_a, \boldsymbol{Y}_a)$, $(\boldsymbol{X}_s, \boldsymbol{Y}_s)$ of the linear matrix inequalities (6.8) and (6.11)
7: Update the Hammerstein-Wiener virtual sensor (6.1) with $\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$ and $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$, and the Hammerstein-Wiener virtual actuator with $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$ and $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_f(0)$
8: Run the reconfigured closed-loop system

**Result:** Globally ISS reconfigured closed-loop system (5.8), (5.4), (6.1), (6.2)

The initialisation problem for the reference system $\Sigma_{\tilde{P}}$ at reconfiguration time can be eased by first updating the observer, and waiting for a certain time period for the observer state to converge towards the true state. After the waiting period, the momentary value of $\hat{\boldsymbol{x}}_f$ may be used for initialising $\tilde{\boldsymbol{x}}$.

**Example 6.2 (Synthesis of Hammerstein-Wiener virtual sensor and virtual actuator for the ship).** *Given the linear elements (4.8), (4.9) of the Hammerstein model, Lemma 6.2 and Lemma 6.3 are applied to the gyro sensor failure $f_1$ and the rudder failure $f_3$.*

*The design of the virtual sensor requires the synthesis of an observer gain $\boldsymbol{L}$ according to Lemma 6.2. In other words, a feasible solution of the LMI (6.8) is sought. The gyro sensor failure is modelled by means of the faulty output matrix $\boldsymbol{C}_f$ defined in Equation (4.16) of Example 4.2. Since the nonlinear input function is a saturation ($\boldsymbol{\varphi} = \mathrm{sat}$), the parameter $\boldsymbol{S}_{\boldsymbol{\varphi}} = \boldsymbol{I}$ is the identity. The numerical search for a solution of the LMI (6.8) leads to the result*

$$\boldsymbol{L} = \begin{pmatrix} 0.32 & 0 & 0 \\ 0.0001 & 0.0001 & 2.724 \\ 0.0001 & 0.0001 & -7.14 \\ -0.0002 & -0.0002 & 97.5 \end{pmatrix}. \tag{6.14}$$

*This result can be interpreted as follows. The yaw rate is represented in the third system state component, and the third column of the matrix $\boldsymbol{L}$ shows that primarily the heading information is used to update the observer yaw rate state. The surge and sway velocities are also used with smaller gains.*

*The design of the virtual actuator amounts to the synthesis of a feedback gain $\boldsymbol{M}$ according to Lemma 6.3. In other words, a feasible solution of the LMI (6.11) is sought. The rudder failure is modelled by means of the faulty input matrix $\boldsymbol{B}_f$ defined in Equation (4.17) of Example 4.2. Since the nonlinear input function is a saturation ($\boldsymbol{\varphi} = \mathrm{sat}$), the parameter $\boldsymbol{S}_{\boldsymbol{\varphi}} = \boldsymbol{I}$ is the identity. The numerical search for a solution of the LMI (6.11) leads to the result*

$$\boldsymbol{M} = 10^{-2} \begin{pmatrix} 1.54 & -0.42 & 0.54 & 0.05 \\ 1.54 & 0.42 & -0.54 & -0.05 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{6.15}$$

*whereas the feedthrough gain $\boldsymbol{N}$ was arbitrarily set to the identity $\boldsymbol{N} = \boldsymbol{I}_3$. The resulting matrix $\boldsymbol{M}$ can be interpreted as follows. The last row is zero, which means that the failed rudder is not used. The first and second rows mean that a too small yaw velocity and too small heading cause the left thruster to increase its force and the right thruster to decrease its force, which causes a right turn and thereby decrease the mismatch between nominal and faulty yaw rate and heading.*

## 6.3 Duality between Hammerstein-Wiener Virtual Sensor and Hammerstein-Wiener Virtual Actuator

The synthesis procedures for the Hammerstein-Wiener virtual sensor (6.1) and the Hammerstein-Wiener virtual actuator (6.2) are closely linked by duality.

**Theorem 6.2 (Duality between the Hammerstein-Wiener virtual sensor and the Hammerstein-Wiener virtual actuator).** *Any solution $\boldsymbol{L}$ to the Hammerstein-Wiener virtual sensor design problem for the pair $(\boldsymbol{C}_f, \boldsymbol{A})$ also parameterises a corresponding solution $\boldsymbol{M}$ to the Hammerstein-Wiener virtual actuator design problem for the pair $(\boldsymbol{A}^T, \boldsymbol{C}_f^T)$. The solutions $\boldsymbol{L}$ and $\boldsymbol{M}$ are linked by means of the relation $\boldsymbol{L} = -\boldsymbol{M}^T$.*

*Proof.* See Appendix D, page 262.

This result means that the duality property linking the linear virtual sensor to the linear virtual actuator is also true in the case of Hammerstein-Wiener systems. The duality holds in the sense that an approach for obtaining suitable gains for one system can be used to obtain suitable gains of the other system.

## 6.4 Special Case: Hammerstein Systems and Actuator Faults

In this section, the special case of actuator faults in Hammerstein systems without output nonlinearity (here given immediately with faulty parameters)

$$\Sigma_{Pf}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{\varphi}_f\left(\boldsymbol{u}_f(t)\right) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t), \end{cases} \tag{6.16}$$

which is obtained from the Hammerstein-Wiener system (5.1) by setting $\boldsymbol{h}(\boldsymbol{y}) = \boldsymbol{y}$, is discussed. The reconfiguration block simplifies from $2n$ dimensions to $n$ dimensions. The robustness of the reconfigured closed-loop system against uncertainties in the model of the faulty system is analysed, and it is shown that the reconfiguration solution is state-feedback universal.

### *Simplified Reconfiguration Block*

In Hammerstein systems (6.16), the measurements are not distorted. Furthermore, the output matrices in the virtual sensor and the virtual actuator are identical due to the assumed absence of sensor faults, and the reference model in the virtual actuator and the model in the observer differ only with respect to the input function $\boldsymbol{\varphi}(\cdot)$ and the input matrix $\boldsymbol{B}$. Therefore, the superposition principle can be applied except for the input parts of the models. After removing the output error injection gain from the virtual sensor, the latter becomes an open-loop predictor that can be merged with the reference model into a single system of the same dimension $n$ as the plant, namely into a Hammerstein virtual actuator, which closely resembles its linear counterpart (4.49) since its state directly represents the difference system, see Fig. 6.4.

**Definition 6.3 (Hammerstein virtual actuator).** The *Hammerstein virtual actuator* is the dynamical system

$$\Sigma_A: \begin{cases} \dot{\boldsymbol{x}}_\Delta(t) &= \boldsymbol{A}\boldsymbol{x}_\Delta(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) - \boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{u}_f(t)) \\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{N}\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{y}_f(t) + \boldsymbol{C}\boldsymbol{x}_\Delta(t) \end{cases} \tag{6.17}$$

with the initial condition $\boldsymbol{x}_\Delta(0) = \boldsymbol{0}$. ◇

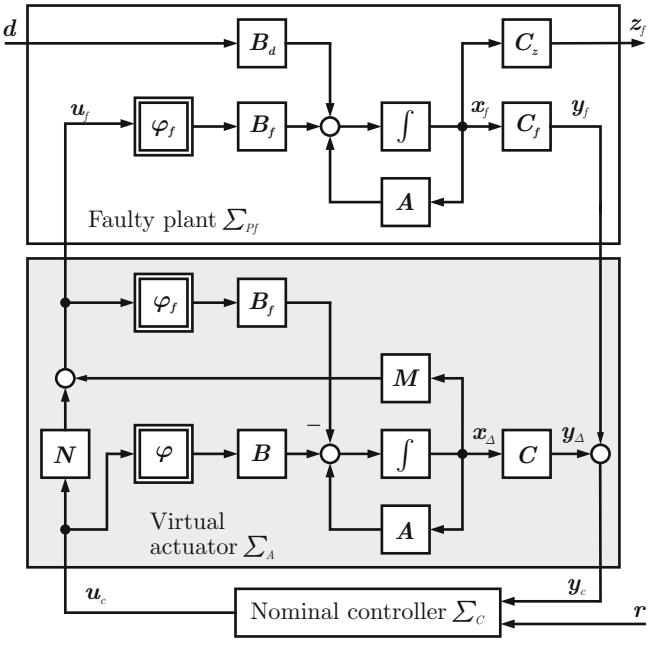


**Fig. 6.4** Reconfiguration block for reconfigurable control after actuator faults in Hammerstein systems.

Due to this simplification, it turns out that the Hammerstein virtual actuator satisfies not only the weak fault-hiding goal, but even the strict fault-hiding goal, as the following Lemma shows.

**Lemma 6.4 (Strict fault-hiding).** *The reconfigured plant $\Sigma_{Pr} = (\Sigma_{Pf}, \Sigma_A)$ that consists of the faulty Hammerstein system (6.16) and the Hammerstein virtual actuator (6.17) satisfies the strict fault hiding goal for arbitrary values of the parameters $\boldsymbol{M}$ and $\boldsymbol{N}$.*

*Proof.* After transformation to the state $\tilde{\boldsymbol{x}} = \boldsymbol{x}_f + \boldsymbol{x}_\Delta$, the reconfigured plant (5.8), (6.17) results in the following transformed reconfigured plant model $\Sigma_{Pr}$:

$$\begin{pmatrix} \dot{\tilde{\boldsymbol{x}}}(t) \\ \dot{\boldsymbol{x}}_\Delta(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}\tilde{\boldsymbol{x}}(t) \\ \boldsymbol{A}\boldsymbol{x}_\Delta(t) - \boldsymbol{B}_f\left(\boldsymbol{\varphi}_f(\boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{N}\boldsymbol{u}_c(t))\right) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{B} \end{pmatrix} \boldsymbol{\varphi}(\boldsymbol{u}_c(t)) + \begin{pmatrix} \boldsymbol{B}_d \\ \boldsymbol{0} \end{pmatrix} \boldsymbol{d}(t) \tag{6.18}$$

$$\begin{pmatrix} \boldsymbol{y}_f(t) \\ \boldsymbol{y}_c(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{C} & -\boldsymbol{C} \\ \boldsymbol{C} & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{x}_\Delta(t) \end{pmatrix}, \begin{pmatrix} \tilde{\boldsymbol{x}}(0) \\ \boldsymbol{x}_\Delta(0) \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_0 \\ \boldsymbol{0} \end{pmatrix}. \tag{6.19}$$

The nominal controller is attached to the system

$$\Sigma_{\tilde{P}} : \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) & = \boldsymbol{A}\tilde{\boldsymbol{x}}(t) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}_c(t) & = \boldsymbol{C}\tilde{\boldsymbol{x}}(t) \end{cases} \tag{6.20}$$

$$\tilde{\boldsymbol{x}}(0) = \boldsymbol{x}_0 + \boldsymbol{x}_{\Delta 0} = \boldsymbol{x}_0$$

by means of the input $\boldsymbol{u}_c$ and the output $\boldsymbol{y}_c$. This system equals the nominal plant model (6.16) up to renaming the state, including the replication of the disturbance effect. It has hence been shown that strict fault-hiding is reached for the initial condition $\boldsymbol{x}_{\Delta 0} = 0$ of the virtual actuator. ■

The difference system is now governed by the dynamics

$$\begin{aligned} \Sigma_\Delta : \begin{cases} \dot{\boldsymbol{x}}_\Delta(t) = \ \boldsymbol{l}(\boldsymbol{x}_\Delta(t), \boldsymbol{u}_c(t)) \\ \boldsymbol{l}(\boldsymbol{x}_\Delta, \boldsymbol{u}_c) \triangleq \boldsymbol{A}\boldsymbol{x}_\Delta + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c) - \boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{M}\boldsymbol{x}_\Delta + \boldsymbol{N}\boldsymbol{u}_c) \\ \boldsymbol{x}_\Delta(0) = \boldsymbol{0} \end{cases} \end{aligned} \tag{6.21}$$

that is simpler than the dynamics of the difference system (6.6).

The conditions under which the Hammerstein virtual actuator stabilises the reconfigured closed-loop system are stated in the following corollary to Theorem 6.1, which gives a solution to Problem 5.1 for Hammerstein systems subject to actuator faults.

**Corollary 6.1 (Reconfigured closed-loop system stability).** *Consider the reconfigured closed-loop system (5.8), (5.4), (6.17) (Fig. 6.4), where the faulty plant has only actuator faults ($\boldsymbol{C}_f = \boldsymbol{C}$, $\boldsymbol{h}_f = \boldsymbol{h}$), and where the diagonal matrix $\boldsymbol{S}_\varphi \in \mathbb{R}^{m\times m}$, $\boldsymbol{S}_\varphi \geq 0$ reflects the sector bounds (5.5) of $\boldsymbol{\varphi}_f(\cdot)$. Suppose that Assumption 5.3 holds and that $\boldsymbol{A}$ is Hurwitz. The reconfigured closed-loop system is ISS w.r.t. its inputs $(\boldsymbol{r}, \boldsymbol{d})$, if there exists a symmetric positive definite matrix $\boldsymbol{X}_a = \boldsymbol{X}_a^T \succ 0$, $\boldsymbol{X}_a \in \mathbb{R}^{n\times n}$, and a matrix $\boldsymbol{Y}_a \in \mathbb{R}^{m\times n}$ such that the LMI (6.11) is satisfied, where $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$.*

In other words, only one LMI has to be solved in order to find the stabilising gain $\boldsymbol{M}$ instead of two.

**Example 6.3 (Hammerstein virtual actuator synthesis for the ship).** *Given the linear elements (4.8), (4.9) of the Hammerstein model, Algorithm 6.1 is applied to the rudder floating failure $f_3$ combined with a reduction of the actuation range $f_4$ for the left thruster. The response of the nonlinear ship model subject to abrupt rudder failure at $t_f = 20$ s is shown in Fig. 6.5. The times $0 \leq t < t_f = 20$ s correspond to the steps 1 to 4 of the algorithm describing the situation without faults, where the virtual actuator is inactive. At the time $t_f = 20$ s, the rudder fails in floating condition, and the left thruster undergoes a reduction of its maximum thrust force to 60% of its nominal force. The fault is immediately detected and isolated in step 4, the model of the faulty system is constructed in step 5, and the virtual actuator parameter $\boldsymbol{M}$ is calculated in step 6, with the result given in Equation (6.15) and brought into effect in step 7. A wind disturbance d appears at $t = 30$ s. Fig. 6.5 shows that the system is indeed stable (the sway velocity approaches the origin for $t > 100$ s). The reduction of the left thruster ($u_1$) maximum force is clearly visible, since its signal is cut off at 0.6. However, the functioning control inputs are hardly used since the matrix $\boldsymbol{M}$ has a very small gain and the setpoints for surge velocity and heading are not reached.*

*Fig. 6.6 shows the motion of the ship with reconfigured controller, which is capable of avoiding the obstacle. However, it does not return to its old course after avoiding the obstacle, therefore the solution is not yet satisfactory.*

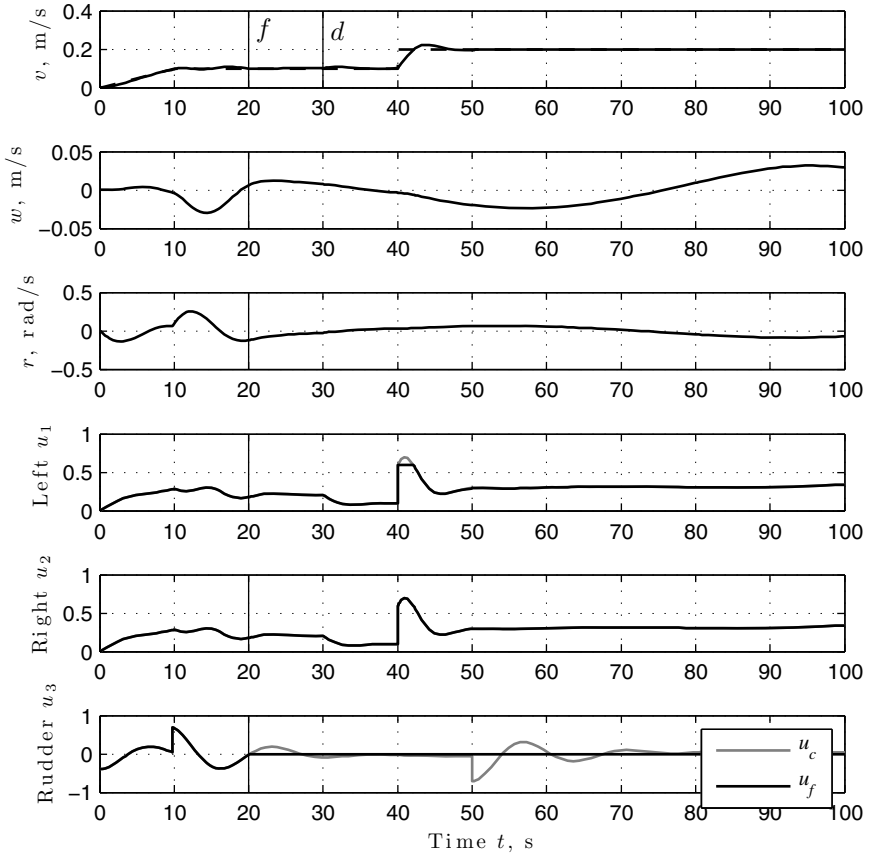


**Fig. 6.5** Response of nonlinear ship subject to rudder failure with control reconfiguration by means of stabilising Hammerstein virtual actuator.

## *Robust Stability*

Suppose that only uncertain models $\hat{\boldsymbol{\varphi}}_f$ and $\hat{\boldsymbol{B}}_f$ of the true faulty nonlinear function $\boldsymbol{\varphi}$ and the true faulty input matrix $\boldsymbol{B}_f$ are available. The uncertainties can be, for example, due to imperfections of the fault diagnosis component. It is shown in this section that, based on the robustness of the nominal closed-loop system, the control reconfiguration approach is robust with respect to bounded uncertainties

$$\delta\boldsymbol{\varphi}_f(\boldsymbol{u}) = \boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{u}) - \hat{\boldsymbol{B}}_f\hat{\boldsymbol{\varphi}}_f(\boldsymbol{u}). \tag{6.22}$$

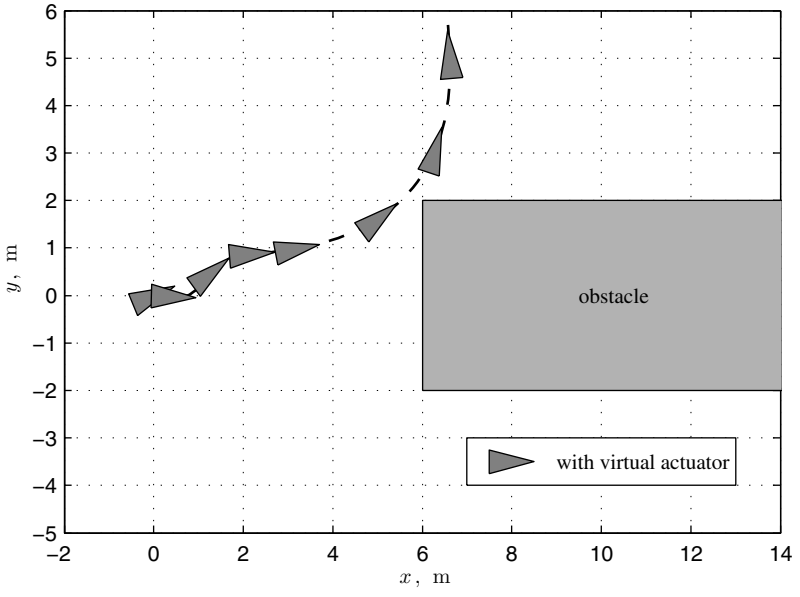


**Fig. 6.6** Motion of nonlinear ship subject to rudder failure with control reconfiguration by means of stabilising Hammerstein virtual actuator.

Consider the following modified model of the transformed reconfigured plant (6.18), (6.19),

$$\Sigma_{\tilde{P}}: \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) = & \boldsymbol{A}\tilde{\boldsymbol{x}}(t) + \left[\boldsymbol{B}_f\boldsymbol{\varphi}_f(\boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t)) - \hat{\boldsymbol{B}}_f\hat{\boldsymbol{\varphi}}_f(\boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t))\right] \\ & + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \end{cases} \tag{6.23}$$

$$\Sigma_{\Delta}: \left\{\dot{\boldsymbol{x}}_{\Delta}(t) = \boldsymbol{A}\boldsymbol{x}_{\Delta}(t) - \hat{\boldsymbol{B}}_f\hat{\boldsymbol{\varphi}}_f(\boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t)) + \boldsymbol{B}\boldsymbol{\varphi}(\boldsymbol{u}_c(t)).\right. \tag{6.24}$$

Clearly, Equation (6.23) shows that the fault-hiding property and the separation property do no longer hold. Consequently, the quantity $\boldsymbol{x}_{\Delta}$ does no longer allow the interpretation as the difference between the nominal and the faulty state. The stability properties of the difference system (6.21)/(6.24), which remains unchanged, however still hold, as long as the sector-bound for $\boldsymbol{\varphi}_f$ is correctly known.

The nonlinear separation principle of cascade-interconnected ISS systems does no longer apply, because the cascade structure is lost. However, Equations (6.23) and (6.24) can be represented as a feedback interconnection of two systems $\Sigma_1$ and $\Sigma_2$ shown in Figure 6.7. Let

$$\|\Sigma_1\|_{\mathcal{L}_2} = \beta_1,$$

and

$$\|\Sigma_2\|_{\mathcal{L}_2} = \beta_2$$

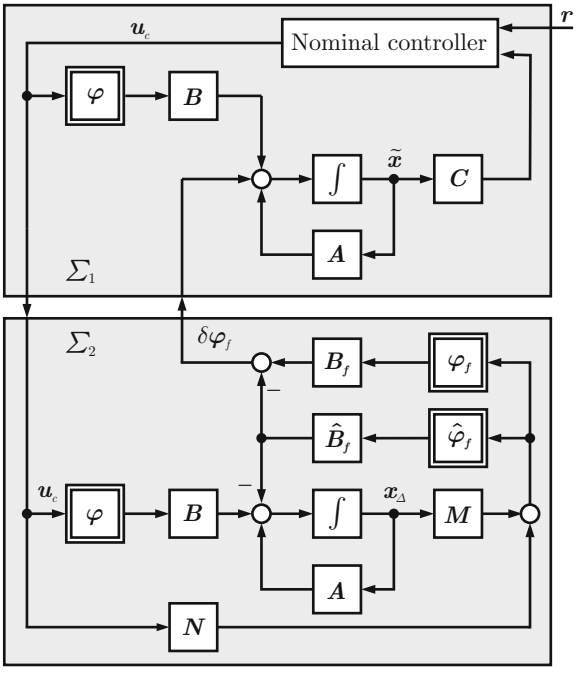


**Fig. 6.7** Redrawn uncertain reconfigured closed-loop Hammerstein system.

denote the $\mathcal{L}_2$ gains of the modified nominal closed-loop system $\Sigma_1$ defined by the nominal plant model (6.23) and the controller (5.4), and the difference system $\Sigma_2$ defined by Equation (6.24). From the small gain theorem 2.3, the following result immediately follows.

**Theorem 6.3 (Robust Hammerstein virtual actuator).** *The reconfigured closed-loop system (5.8), (5.4), (6.17) is robustly stable against uncertainties (6.22) in the nonlinear function $\boldsymbol{\varphi}_f$, if $\beta_1\beta_2 < 1$ holds.*

Assuming that the gain of the nominal closed-loop system ($\Sigma_1$, upper block in Fig. 6.7) from the fictitious disturbance input $\delta\boldsymbol{\varphi}_f$ to the control input $\boldsymbol{u}_c$ is finite, the uncertain loop is always stable if the model uncertainties $\delta\boldsymbol{\varphi}_f$ are sufficiently small. This insight is gained from consideration of $\Sigma_2$ (see Fig. 6.7) and by recognising the fact that $\delta\boldsymbol{\varphi}_f$ is the output of $\Sigma_2$. This theorem shows the robustness of the concept of the virtual actuator with respect to model uncertainties.

*Remark 6.2 (Relation to nominal robustness).* It seems that the robust stability of the reconfigured closed-loop system does not depend on the nominal robust stability. This surprising result is due to the fact that the Hammerstein virtual actuator

implicitly introduces the nominal system model with the model function $\hat{\boldsymbol{\varphi}}$ as a reference model. Its relationship to the true unknown function $\boldsymbol{\varphi}$ does not appear. On the other hand, the system $\Sigma_2$ in Fig. 6.7 may be interpreted as a parallel uncertainty of the input path in the nominal system. With this interpretation, the robustness of the reconfigured closed-loop system against uncertainties in $\boldsymbol{B}_f$ and $\boldsymbol{\varphi}_f$ requires a certain level of nominal robustness against uncertainty in $\boldsymbol{B}$ and $\boldsymbol{\varphi}$. ∘

### *State-Feedback Universality*

In this section, it is shown that the choice of the fault-hiding approach in general and the Hammerstein virtual actuator in particular are not restrictive for solving the stability recovery problem for Hammerstein systems subject to actuator faults. First, the notion of state-feedback universality is defined.

**Definition 6.4 (Linear state-feedback universality).** A reconfigurable control scheme $S$ is called *linear state-feedback universal* for the class of Hammerstein systems subject to actuator faults, if there exists no stabilising linear state-feedback controller $\boldsymbol{u}_f(t) = \boldsymbol{K}\boldsymbol{x}_f(t)$ for the faulty plant (5.8) whenever the plant (5.8) with actuator faults is not stabilisable with $S$. ◇

**Theorem 6.4 (Linear state-feedback universal Hammerstein virtual actuator).** *The reconfigurable control scheme $\Sigma_{Cr} = (\Sigma_A, \Sigma_C)$ defined by the Equations (5.4), (6.17) with the stabilisability condition (6.11) is linear state-feedback universal for Hammerstein systems subject to actuator faults.*

*Proof.* See Appendix D, page 262.

The proof shows that the problems of linear state-feedback synthesis and virtual actuator synthesis are equivalent. The proof does not depend on the chosen stabilisation technique and is therefore not conservative. It should be stressed that the universality statement is limited to linear state feedback. The application of nonlinear state-feedback control eases the stabilisation task if the input characteristic $\boldsymbol{\varphi}$ is invertible. However, saturation functions do not admit a global inverse.

## 6.5 Special Case: Hammerstein-Wiener Systems and Sensor Faults

### *Simplified Reconfiguration Block*

In the case of pure sensor faults, the Hammerstein-Wiener virtual sensor $\Sigma_S$ defined in Equation (6.1) is used with the simplifications $\boldsymbol{B}_f = \boldsymbol{B}$ and $\boldsymbol{\varphi}_f(\cdot) = \boldsymbol{\varphi}(\cdot)$ (see Fig. 6.8).

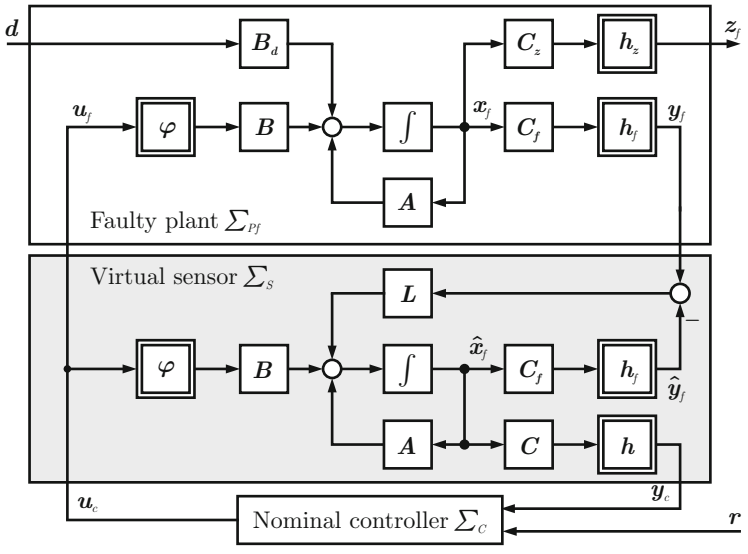


**Fig. 6.8** Reconfiguration block for reconfigurable control after sensor faults in Hammerstein-Wiener systems.

**Lemma 6.5 (Asymptotic fault-hiding).** *The reconfigured plant* $(\Sigma_{Pf}, \Sigma_S)$ *that consists of the faulty Hammerstein-Wiener system (5.8) with sensor faults* ($\boldsymbol{B}_f = \boldsymbol{B}, \boldsymbol{\varphi}_f(\cdot) = \boldsymbol{\varphi}(\cdot)$) *and the Hammerstein-Wiener virtual sensor (6.1) with* $\boldsymbol{B}_f = \boldsymbol{B}$ *and* $\boldsymbol{\varphi}_f(\cdot) = \boldsymbol{\varphi}(\cdot)$ *satisfies the asymptotic fault hiding goal for all choices of* $\boldsymbol{L}$ *that make the observation error globally asymptotically stable for* $\boldsymbol{d} \equiv \boldsymbol{0}$.

*Proof.* After transformation to the observation error $\boldsymbol{e} = \hat{\boldsymbol{x}}_f - \boldsymbol{x}_f$, the dynamics of the reconfigured plant is given by the equations

$$\begin{pmatrix} \dot{\boldsymbol{x}}_f(t) \\ \dot{\boldsymbol{e}}(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{A} \end{pmatrix} \begin{pmatrix} \boldsymbol{x}_f(t) \\ \boldsymbol{e}(t) \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{L} \end{pmatrix} \Big( \boldsymbol{h}_f(\boldsymbol{C}_f \boldsymbol{x}_f(t)) - \boldsymbol{h}_f(\boldsymbol{C}_f(\boldsymbol{x}_f(t) + \boldsymbol{e}(t)) \Big) \tag{6.25}$$

$$+ \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{0} \end{pmatrix} \boldsymbol{\varphi}(\boldsymbol{u}_c(t)) + \begin{pmatrix} \boldsymbol{B}_d \\ -\boldsymbol{B}_d \end{pmatrix} \boldsymbol{d}(t), \begin{pmatrix} \boldsymbol{x}_f(0) \\ \boldsymbol{e}(0) \end{pmatrix} = \begin{pmatrix} \boldsymbol{x}_0 \\ \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0 \end{pmatrix} \tag{6.26}$$

$$\begin{pmatrix} \boldsymbol{y}_f(t) \\ \boldsymbol{y}_c(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{h}_f(\boldsymbol{C}_f \boldsymbol{x}_f(t)) \\ \boldsymbol{h}(\boldsymbol{C}(\boldsymbol{x}_f(t) + \boldsymbol{e}(t))) \end{pmatrix}. \tag{6.27}$$

Clearly, the controller connected to the output $\boldsymbol{y}_c$ sees entirely nominal dynamics for $\boldsymbol{e}(t) = 0$, which is asymptotically achieved for globally asymptotically stable observation error dynamics. ■

The following corollary to Theorem 6.1 provides sufficient conditions on $\boldsymbol{L}$ which guarantee that the observation error is globally asymptotically stable for $\boldsymbol{d} \equiv \boldsymbol{0}$ and ISS w.r.t. $\boldsymbol{d}$.

**Corollary 6.2 (Reconfigured closed-loop system stability).** *Consider the reconfigured closed-loop system (5.8), (5.4), (6.1) (Fig. 6.8), where the faulty plant has only sensor faults ($\boldsymbol{B}_f = \boldsymbol{B}$, $\boldsymbol{\varphi}_f = \boldsymbol{\varphi}$), and where the diagonal matrix $\boldsymbol{S}_h \in \mathbb{R}^{m\times m}$, $\boldsymbol{S}_h \geq 0$ reflects the sector bounds (5.6) of $\boldsymbol{h}_f(\cdot)$. Suppose that Assumption 5.3 holds, that $\boldsymbol{A}$ is Hurwitz. The reconfigured closed-loop system is ISS w.r.t. its input $(\boldsymbol{r}, \boldsymbol{d})$, if there exists a symmetric positive definite matrix $\boldsymbol{X}_s = \boldsymbol{X}_s^T > 0$, $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$, and a matrix $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$ such that the LMI (6.8) is satisfied, where $\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$.*

In other words, only one LMI has to be solved to find the stabilising gain $\boldsymbol{L}$ instead of two.

**Example 6.4 (Hammerstein-Wiener virtual sensor synthesis for the ship).** *Given the elements (4.8), (4.9) of the linear model, Algorithm 6.1 is applied to the gyro sensor failure $f_1$. The response of the nonlinear ship model subject to abrupt gyro sensor failure at $t_f = 20$ s is shown in Fig. 6.9. The times $0 \leq t < t_f = 20$ s correspond to the steps 1 to 4 of the algorithm describing the situation without faults, where the virtual sensor is inactive. At the time $t_f = 20$ s, the gyro sensor fails. The fault is immediately detected and isolated in step 4, the model of the faulty system*

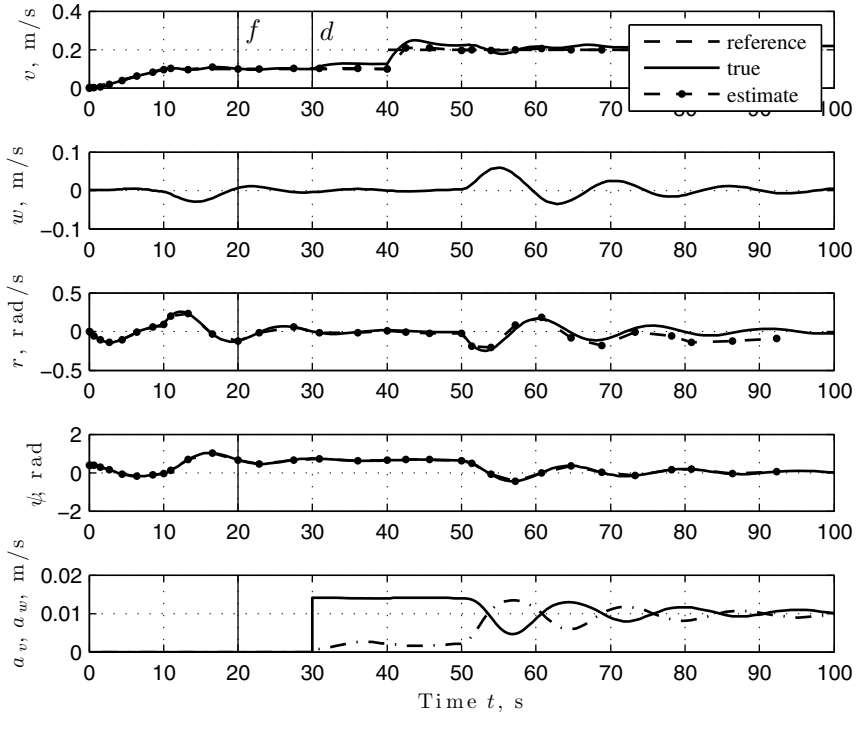


**Fig. 6.9** Response of nonlinear ship subject to gyro sensor failure with control reconfiguration by means of stabilising Hammerstein-Wiener virtual sensor.

*is constructed in step 5, and the virtual sensor parameter* $\boldsymbol{L}$ *is calculated in step 6, with the result given in Equation (6.14), and brought into effect in step 7.*

*Fig. 6.9 shows the true surge, sway, and yaw velocities as well as the estimates for surge and yaw velocity. The figure demonstrates that the virtual sensor provides good estimates that permit the controller to steer the ship away from the obstacle, as Fig. 6.10 clearly shows. However, the wind disturbance becoming effective at* $t = 30$ *s cannot be compensated, and the state estimate deviates from the true values. The stronger the wind is, the larger the observation error becomes.*

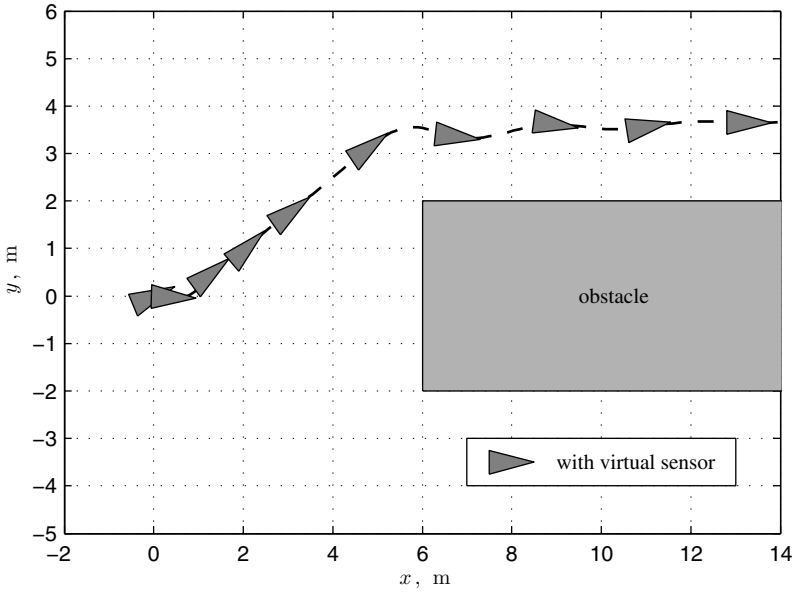


**Fig. 6.10** Motion of nonlinear ship subject to gyro sensor failure with control reconfiguration by means of stabilising Hammerstein-Wiener virtual sensor.

## *Robust Stability*

Suppose that the output model of the Hammerstein-Wiener system is subject to uncertainties of the nonlinear output function

$$\delta \boldsymbol{h}_f(\boldsymbol{x}) = \boldsymbol{h}_f(\boldsymbol{C}_f \boldsymbol{x}) - \hat{\boldsymbol{h}}_f(\hat{\boldsymbol{C}}_f \boldsymbol{x}). \tag{6.28}$$

Such an uncertainties are likely to arise due to imperfections of the fault diagnosis component. The uncertainties modify the dynamics of the observation error (6.5) for the case of pure sensor faults to

$$\begin{aligned} \Sigma_e : \left\{ \dot{\boldsymbol{e}}(t) \right. &= \boldsymbol{A}\boldsymbol{e}(t) + \boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f \boldsymbol{x}_f(t)) - \hat{\boldsymbol{h}}_f(\hat{\boldsymbol{C}}_f(\boldsymbol{x}_f(t) + \boldsymbol{e}(t)))\Big) - \boldsymbol{B}_d \boldsymbol{d}(t) \\ \boldsymbol{e}(0) &= \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0. \end{aligned} \tag{6.29}$$

The following theorem states that the closed-loop stability properties do not change in the case of model uncertainties, if the uncertain output function satisfies a robust Lipschitz condition.

**Theorem 6.5 (Robust Hammerstein-Wiener virtual sensor).** *The reconfigured closed-loop system (5.8), (5.4), (6.1) subject to sensor faults is robustly stable against uncertainties (6.28) in the nonlinear function $\boldsymbol{h}_f$ and the output matrix $\boldsymbol{C}_f$, if there exists numbers $\hat{R}_0 > 0$ and $\hat{R} > 0$ such that the uncertainties satisfy the robust Lipschitz condition*

$$\forall \boldsymbol{x}, \boldsymbol{e} : \ \|\boldsymbol{h}_f(\boldsymbol{C}_f \boldsymbol{x}) - \hat{\boldsymbol{h}}_f(\hat{\boldsymbol{C}}_f(\boldsymbol{x}+\boldsymbol{e}))\| \leq \hat{R}_0 + \hat{R}\|\boldsymbol{e}\|. \tag{6.30}$$

The proof closely follows the proof of Lemma 6.2.

*Proof.* See Appendix D, page 262.

Due to the robustness property, modeling errors in the output matrix and nonlinear output function made by the diagnosis component are tolerated by the virtual sensor with respect to stability. The larger the uncertainties are, the larger the persistent observation error is.

## *Output-Injection Universality*

In this section, it is shown that the choice of the fault-hiding approach in general and the Hammerstein-Wiener virtual sensor in particular are not restrictive for solving the stability recovery problem for Hammerstein-Wiener systems subject to sensor faults. The notion of output-injection universality is defined as follows.

**Definition 6.5 (Output-injection universality).** A reconfigurable control scheme $S$ is called *output-injection universal* for the class of Hammerstein-Wiener systems subject to sensor faults, if there exists no stabilising linear output-injection controller $\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{K}\boldsymbol{y}_f(t)$ for the faulty plant (5.8) whenever the faulty plant (5.8) is not stabilisable with $S$. ◇

**Theorem 6.6 (Output-injection universal Hammerstein-Wiener virtual sensor).** *The reconfigurable control scheme $\Sigma_{Cr} = (\Sigma_S, \Sigma_C)$ defined by the Equations (5.4), (6.1) with the stabilisability condition (6.8) is output-injection universal for Hammerstein-Wiener systems subject to sensor faults.*

*Proof.* See Appendix D, page 263.

The proof is conducted by showing that the problems of output-injection synthesis and virtual sensor synthesis are equivalent. As previously in the case of state-feedback universality of virtual actuators, the proof does not depend on the chosen stabilisation technique and is therefore not conservative.

## 6.6 Summary and Discussion

In this chapter, the stability recovery problem has been solved for the class of Hammerstein-Wiener systems subject to actuator and sensor faults modelled by means of modified input and output matrices and modified respective nonlinear functions (Theorem 6.1). A synthesis procedure has been given to determine stabilising gains of the Hammerstein-Wiener virtual sensor and the Hammerstein-Wiener virtual actuator (Algorithm 6.1). Two interesting special cases have been separately analysed: actuator faults in Hammerstein systems and sensor faults in Hammerstein-Wiener systems.

It has been shown that the concepts of Hammerstein virtual actuators and Hammerstein-Wiener virtual sensors are robust with respect to modelling errors in the faulty nonlinear input and output functions (Theorem 6.3, Theorem 6.5). Furthermore, it was shown through universality properties that the choices of virtual sensors and virtual actuators as reconfiguration approaches are not restrictive (Theorem 6.4, Theorem 6.6). The universality results are independent of the synthesis techniques for virtual actuators and state-feedback controllers on the one hand and virtual sensors and output-injection controllers on the other hand. Therefore, any novel state-feedback synthesis technique that is less conservative than the absolute stability technique chosen in this chapter can also be applied to the synthesis of virtual actuator gains. A similar statement is true for the problems of virtual sensor and output injection gain synthesis.

Clearly, the robustness properties derived for the Hammerstein virtual actuator and the Hammerstein-Wiener virtual sensor also apply to their linear counterparts, which are special cases of the nonlinear versions considered in this chapter. The found stability robustness against uncertainties in the model of the faulty plant is crucial, since the models coming from fault diagnosis components are often considerably more uncertain than models derived from first principles modelling, parameter identification, or black-box identification with the engineer in the synthesis loop.

The synthesis problems of finding stabilising gains for the Hammerstein-Wiener virtual actuator and virtual sensor turn out to be dual. However, the symmetry between the virtual actuator and sensor also has limitations, as shown in the robustness analysis. The Hammerstein virtual actuator is small-gain robust, whereas no small-gain condition was needed for the virtual sensor. The broken symmetry is explained by the different entities processed by the virtual actuator and virtual sensor: information in the case of the virtual sensor, and energy in the case of the virtual actuator. The rerouting of control energy to adequate actuators is unavoidable for control, which explains the small-gain condition. Furthermore, the measured output

provided to the nominal controller is entirely based on the state estimate in case of the Hammerstein-Wiener virtual sensor. The faulty measured output is not directly fed through.

With respect to the requirements stated in Chapter 1.3, the method is suitable for autonomous online application, provided that the stated synthesis LMIs can be autonomously solved. Given the current computational power, it is numerically applicable to small and medium sized problems. The established robustness properties are particularly important to safeguard against inaccurate models derived by the fault diagnosis component.

This chapter has entirely focused on the recovery of stability. The ship example has, however, well demonstrated that stability alone is not sufficient for applicable reconfiguration solution. The following two chapters thus discuss the recovery of asymptotic setpoint tracking and the recovery of nominal performance properties.

# Chapter 7
# Setpoint Tracking Recovery after Actuator Faults in Saturated Systems

**Abstract.** This chapter gives the solution to the tracking recovery problem after actuator faults in saturated systems. The solution extends the stabilising reconfiguration solution to achieve tracking of constant setpoints. Feasible setpoints that can be reached by the faulty plant are characterised, and it is shown that a suitably designed saturated virtual actuator ensures that they be reached. It is furthermore shown how infeasible setpoints are mapped to feasible ones.

## 7.1 The Setpoint Tracking Problem

This chapter describes a solution to the constant setpoint tracking recovery problem 5.2 as an extension to the stability recovery solution of the previous chapter. The extension is described for the class of saturated systems subject to actuator faults, which is a special class of Hammerstein systems where the nonlinear input function is a saturation function.

The tracking problem is affected by faults in two ways:

- Actuator failures typically reduce the available degrees of freedom for independently controlling all outputs, such that after the fault, an output vector with reduced dimension is independently assignable.
- Even if the tracking output dimensions may remain unchanged, the reachable set in the output space is generally reduced due to the fault, for example, because of smaller actuation ranges.

This chapter is entirely based on the following assumption.

**Assumption 7.1 (Input saturation functions).** *The nonlinear functions* $\boldsymbol{\varphi}(\cdot)$ *and* $\boldsymbol{\varphi}_f(\cdot)$ *of the systems (5.1), (5.8) are pure saturations* $\boldsymbol{\varphi}(\boldsymbol{u}) = sat(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u})$, *and* $\boldsymbol{\varphi}_f(\boldsymbol{u}_f) = \mathrm{sat}(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f)$.

The assumption includes tightened saturation bounds caused by actuator faults, but it excludes changes of the slope in the sensitivity range. Between saturation limits, the nominal and faulty saturation functions must be identities.

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 119–130.
springerlink.com 

The class of systems considered in this chapter is thus the class of nominal saturated systems

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\,\mathrm{sat}\left(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u}_c(t)\right) + \boldsymbol{B_d}\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t) \end{cases} \tag{7.1}$$
$$\boldsymbol{x}(0) = \boldsymbol{x}_0,$$

first defined above in Definition (5.2). The class of saturated systems subject to actuator faults is

$$\Sigma_{Pf} : \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\,\mathrm{sat}\left(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f(t)\right) + \boldsymbol{B_d}\boldsymbol{d}(t) \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t) \end{cases} \tag{7.2}$$
$$\boldsymbol{x}_f(0) = \boldsymbol{x}_0,$$

first defined above in Equation (5.9). The following property of the nominal and faulty saturated plants is assumed.

**Assumption 7.2 (Finite static gain).** *The linear part of the nominal saturated system (7.1) and the linear part of the faulty saturated system (7.2) have finite static gains from the saturated control input to the relevant outputs $\boldsymbol{z}$ and $\boldsymbol{z}_f$:*

$$\|\boldsymbol{C}_z\boldsymbol{A}^{-1}\boldsymbol{B}\| < \infty \tag{7.3}$$
$$\|\boldsymbol{C}_z\boldsymbol{A}^{-1}\boldsymbol{B}_f\| < \infty. \tag{7.4}$$

Throughout this chapter, the Hammerstein virtual actuator is specialised as a saturated virtual actuator defined as follows (Fig. 7.1).

**Definition 7.1 (Saturated virtual actuator).** The *saturated virtual actuator* is the dynamical system

$$\Sigma_A : \begin{cases} \dot{\boldsymbol{x}}_{\Delta}(t) &= \boldsymbol{A}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{B}\,\mathrm{sat}\left(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \boldsymbol{u}_c(t)\right) - \boldsymbol{B}_f\,\mathrm{sat}\left(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{u}_f(t)\right) \\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{N}\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{y}_f(t) + \boldsymbol{C}\boldsymbol{x}_{\Delta}(t) \end{cases} \tag{7.5}$$

with the initial condition $\boldsymbol{x}_{\Delta}(0) = \boldsymbol{0}$. ◇

The tracking recovery problem is restricted to the relevant outputs $\boldsymbol{z}$ and $\boldsymbol{z}_f$ of the nominal and faulty systems (7.1) and (7.2). The dimension of independently assignable outputs $\dim(\boldsymbol{z}_f)$ depends on the remaining degrees of freedom after the fault. With

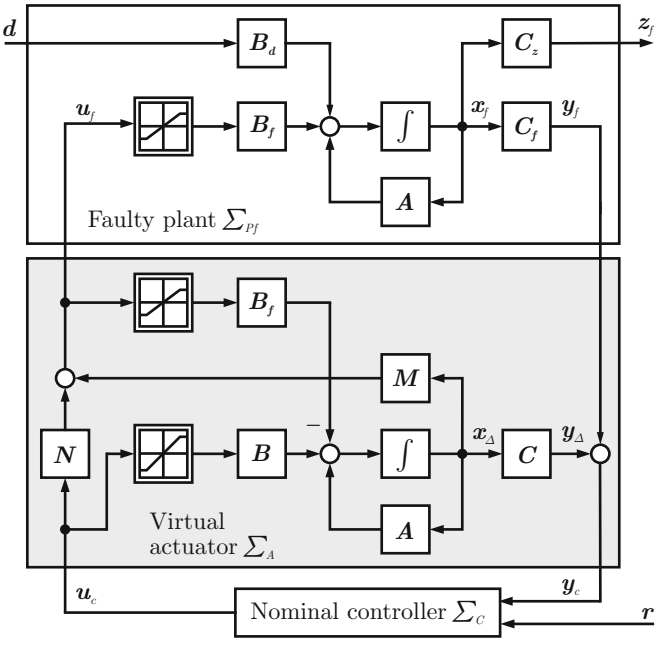


**Fig. 7.1** Reconfiguration block for reconfigurable control after actuator faults in saturated systems.

these assumptions, the reconfiguration problem solved in this chapter consists in solving Problem 5.2, which may be reformulated as follows.

**Lemma 7.1 (Setpoint tracking recovering saturated virtual actuator).** *Given a stabilising gain* $\boldsymbol{M}$ *according to Corollary 6.1, Problem 5.2 is equivalent to finding the gain* $\boldsymbol{N}$ *of the saturated virtual actuator (7.5) such that*

$$\forall \boldsymbol{u}_c(t) = \bar{\boldsymbol{u}}_c \rho(t) : \lim_{t\to\infty} z(t) - z_f(t) = \boldsymbol{0}. \tag{7.6}$$

In other words, the output $z_\Delta(t) \triangleq z(t) - z_f(t) = \boldsymbol{C}_z \boldsymbol{x}(t) - \boldsymbol{C}_z \boldsymbol{x}_f(t) = \boldsymbol{C}_z \boldsymbol{x}_\Delta(t)$ should be statically decoupled from the input $\boldsymbol{u}_c(t)$.

The solution for saturated systems is organised as follows. The solution is prepared by defining several relevant polytopes and their relations in Section 7.2. The main result regarding the recovery of nominal tracking properties is stated in Section 7.3. A setpoint supervision approach that maps infeasible equilibria to feasible ones is described in Section 7.4.

## 7.2 Concept for Setpoint Tracking Recovery

The solution to Problem 5.2 starts with the solution to the linear problem. The basic idea for extending the linear solution to saturated systems consists in describing the

saturations as polytopes restricting the input space, instead of describing them as nonlinear functions in the input path. Through this idea, the saturation operator is replaced by an intersection operator.

The notions of the sets of *feasible control inputs* and *feasible outputs* are central to the present approach. All these sets are represented by convex polytopes, which are defined in this section, after recalling the linear case.

In the linear case, the tracking problem for constant setpoint concerns the transfer function

$$\boldsymbol{T}_{\boldsymbol{u}_c \to z_\Delta}(s) = \boldsymbol{C}_z(s\boldsymbol{I} - (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M}))^{-1}(\boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}). \tag{7.7}$$

Its solution for given $\boldsymbol{M}$ amounts to the selection of a matrix $\boldsymbol{N}$ such that the transfer function $\boldsymbol{T}_{\boldsymbol{u}_c \to z_\Delta}(s)$ has a zero at the frequency $s = 0$: $\boldsymbol{T}_{\boldsymbol{u}_c \to z_\Delta}(0) = \boldsymbol{0}$. This goal is known to be achievable for linear systems ($\boldsymbol{\varphi}(\boldsymbol{u}_c) = \boldsymbol{u}_c$) if and only if the condition (4.54) is satisfied (see Chapter 4). The condition (4.54) limits the number of independently assignable output components of $\boldsymbol{z}$ to $\dim(\boldsymbol{z}) = p \leq \operatorname{rank}(\boldsymbol{C}_z\boldsymbol{A}^{-1}\boldsymbol{B}_f)$. The corresponding solution $\boldsymbol{N}$ is obtained from the equation

$$\boldsymbol{N} = \left(\boldsymbol{C}_z(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}_f\right)^{\dagger}\boldsymbol{C}_z(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})^{-1}\boldsymbol{B}. \tag{7.8}$$

The linear virtual actuator (4.49) also transforms the control command $\boldsymbol{u}_c$ into the input $\boldsymbol{u}_f$ of the faulty plant. These signals are connected by means of the transfer function

$$\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(s) = \boldsymbol{M}(s\boldsymbol{I} - (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M}))^{-1}(\boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}) + \boldsymbol{N}$$

with the static gain

$$\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(0) = \boldsymbol{N} - \boldsymbol{M}(\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})^{-1}(\boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}). \tag{7.9}$$

**Definition 7.2 (Sets of feasible and generated control inputs).** Consider the saturated system (7.2) with actuator faults. The *set of feasible control inputs* is defined as

$$\mathcal{U} \triangleq \{\boldsymbol{u} \in \mathbb{R}^m : \underline{u}_{f,i} \leq u_i \leq \overline{u}_{f,i},\ i = 1, \ldots, m\}. \tag{7.10}$$

The set of *generated control inputs* is defined as

$$\mathcal{U}_a \triangleq \{\boldsymbol{u} \in \mathcal{U} : \exists \boldsymbol{y}(t), \boldsymbol{r}(t), \boldsymbol{x}_c(t), \boldsymbol{x}_{c0}, t \text{ such that } (\boldsymbol{y}, \boldsymbol{r}, \boldsymbol{x}_c, \boldsymbol{u}_c) \text{ satisfies (5.4)}\}. \tag{7.11}$$

$\diamond$

The set of feasible control inputs $\mathcal{U}$ thus describes all inputs allowed by the saturations. The set of generated control inputs $\mathcal{U}_a$ is the subset of $\mathcal{U}$ describing all inputs that are actually generated by the controller. If the controller ignores entire components of the input vector, then $\mathcal{U}_a$ is a proper subset ($\mathcal{U}_a \subset \mathcal{U}$). By definition, all

previously defined sets are described by convex polytopes. The *set of safe outputs* $\mathcal{Z}_s \subseteq \mathbb{R}^p$ contains all output values that do not violate safety constraints.

This set is defined by the plant designer and is assumed to be convex as well. A constant feasible control input $\boldsymbol{u}_c$ applied to the reconfigured plant $\Sigma_{Pr} = (\Sigma_{Pf}, \Sigma_A)$ does not activate the saturations in the saturated virtual actuator, where $\Sigma_{Pf}$ is a saturated system subject to actuator faults, and $\Sigma_A$ is a saturated virtual actuator. In steady-state, a constant feasible input $\bar{\boldsymbol{u}}_c$ is transformed into a constant steady-state input $\bar{\boldsymbol{u}}_f$ to the faulty plant by means of the static gain (7.9):

$$\bar{\boldsymbol{u}}_f = \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(0)\bar{\boldsymbol{u}}_c.$$

The set $\mathcal{U}_a$ is mapped by the virtual actuator to the set

$$\mathcal{U}_f = \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(0)\mathcal{U}_a. \tag{7.12}$$

Since the control input to the faulty plant is limited by its saturations, the effective set of control inputs behind the saturations is

$$\mathcal{U}_s = \mathcal{U}_f \cap \mathcal{U}. \tag{7.13}$$

One may now ask which original control inputs from the set $\mathcal{U}_a$ translate into effective control inputs in the set $\mathcal{U}_s$. This set of saturation-respecting control inputs is called $\mathcal{U}_c$ and defined as

$$\mathcal{U}_c = \{\boldsymbol{u} \in \mathcal{U}_a : \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(0)\boldsymbol{u} \in \mathcal{U}_s\}$$

and calculated from the formula

$$\mathcal{U}_c = \boldsymbol{T}^{+}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(0)\mathcal{U}_s.$$

The relationship between the sets $\mathcal{U}$, $\mathcal{U}_a$, $\mathcal{U}_f$, and $\mathcal{U}_s$ is illustrated in Fig. 7.2.

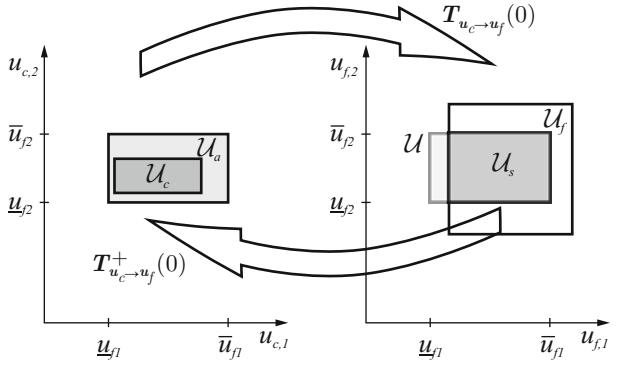


**Fig. 7.2** Hypercube of admissible inputs defined by input saturations.

From these sets and Assumption 7.2, it is possible to calculate the admissible output equilibria

$$\mathcal{Z}_a \triangleq (\boldsymbol{C}_z \boldsymbol{A}^{-1} \boldsymbol{B}_f \mathcal{U}_s) \cap \mathcal{Z}_s. \tag{7.14}$$

The elements of $\mathcal{Z}_a$ are output equilibria that exist for the saturated system with actuator faults.

The final concept needed for the statement of the main result is the concept of a minimum equilibrium-preserving input matrix.

**Definition 7.3 (Minimum equilibrium-preserving input matrix).** An input matrix $\boldsymbol{B}_f$ of a faulty linear dynamical system (4.14) is called *minimum equilibrium-preserving*, if the quadruple $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{B}_f, \boldsymbol{C}_z)$ satisfies Condition (4.54), and if for every matrix $\hat{\boldsymbol{B}}_f$ obtained from $\boldsymbol{B}_f$ by setting some column of $\boldsymbol{B}_f$ to zero implies that the quadruple $(\boldsymbol{A}, \boldsymbol{B}, \hat{\boldsymbol{B}}_f, \boldsymbol{C}_z)$ does not satisfy Condition (4.54) any longer. ⋄

The minimum equilibrium-preserving condition on the input matrix prevents the existence of multiple input equilibria leading to the same output equilibrium, of which one might activate saturations and be infeasible, whereas the other might be feasible.

## 7.3 Main Setpoint Tracking Recovery Result

The following theorem states that the linear solution approach to the tracking recovery problem also solves the tracking recovery problem for saturated systems, if the requested setpoint is feasible, and if an additional technical condition on the input matrix is satisfied. The following theorem thus gives the solution to Problem 5.2.

**Theorem 7.1 (Stable asymptotic setpoint tracking recovery).** *Consider the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_A, \Sigma_C)$ that consists of the faulty saturated plant (7.2), of the saturated virtual actuator (7.5), and of the nominal controller (5.4), where the gain $\boldsymbol{M}$ is obtained from Corollary 6.1, and where the gain $\boldsymbol{N}$ is obtained from Equation (7.8). Then, any feasible constant setpoint $\bar{\boldsymbol{r}} \in \mathcal{Z}_a$ is tracked by the reconfigured closed-loop system to the same precision as it is tracked by the nominal closed-loop system (5.4), (7.1),*

$$\lim_{t \to \infty} \boldsymbol{z}(t) - \boldsymbol{z}_f(t) = \boldsymbol{0},$$

*if the input matrix $\boldsymbol{B}_f$ of the faulty saturated system is minimum equilibrium-preserving.*

*Proof.* See Appendix D, page 263.

The reason for this result is mainly the absence of obstacles in the state space due to the presence of simple saturation functions. The saturation functions imply convexity of all relevant polytopes, therefore control inputs generated by continuous feedback control laws between two feasible equilibria are always feasible in between due to convexity. The presence of state-space obstacles leads to the class of nonholonomic systems, which can generally not be controlled by means of continuous inputs [104].

It is a hypothesis of Theorem 7.1 that a given reference setpoint $\bar{\boldsymbol{r}}$ is feasible, and likewise that the input matrix $\boldsymbol{B}_f$ is minimum equilibrium-preserving. Both hypothesis may not be directly satisfied and must be enforced by setpoint supervision and input matrix reduction, which are described in the next section.

## 7.4 Setpoint Supervision and Input Matrix Reduction

Reference inputs that are externally provided by fault-unaware components need not necessarily be feasible. The set of feasible output equilibria $\mathcal{Z}_a$ for the faulty system may be strictly smaller than the nominal one. Demanding infeasible setpoints often leads to the activation of actuator saturations, with negative consequences for the closed-loop performance. It is thus reasonable to demand only achievable setpoints by mapping infeasible reference inputs to feasible ones (Fig. 7.3).

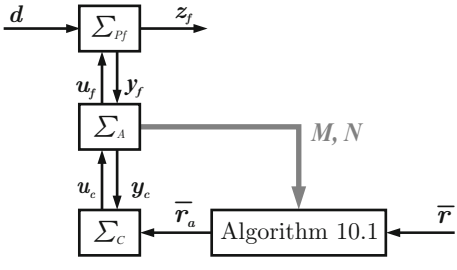


**Fig. 7.3** Integration of setpoint projection into overall reconfigurable control scheme.

To obtain a feasible setpoint $\bar{\boldsymbol{r}}_a \in \mathcal{Z}_a$ from an arbitrary setpoint $\bar{\boldsymbol{r}}$, $\bar{\boldsymbol{r}}$ is projected onto the set $\mathcal{Z}_a$. The projection may be

- orthogonal, or
- along the set of relevant outputs $\boldsymbol{z}$.

The first approach potentially changes all elements of $\bar{\boldsymbol{r}}$, while the second approach preserves as many setpoint elements of $\bar{\boldsymbol{r}}$ as possible. In particular, the number $\kappa$ of outputs whose setpoint is independently assignable is

$$\kappa = \operatorname{rank}\left(\boldsymbol{C}_z \boldsymbol{A}^{-1} \boldsymbol{B}_f\right). \tag{7.15}$$

The relevant output space basis $\boldsymbol{T}_z \in \mathbb{R}^{q\times q}$ marks $\kappa$ relevant elements of the outputs $z$ by means of ones. Assuming without loss of generality that the first $\kappa$ outputs are chosen (always achievable by permutation of outputs), the basis is given by the matrix

$$\boldsymbol{T}_z = \begin{pmatrix} \boldsymbol{I}_\kappa & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}, \tag{7.16}$$

where the zero blocks have appropriate dimensions. To assure that the equilibrium is met for the components $\bar{\boldsymbol{r}}_z$ of $\bar{\boldsymbol{r}}$ marked in $\boldsymbol{T}_z$, the polytope $\mathcal{Z}_a$ represented by $\boldsymbol{H}_a$ and $\boldsymbol{k}_a$ is augmented by inequalities that constrain the polytope to $\bar{\boldsymbol{r}}_z$,

$$\boldsymbol{H}_p = \begin{pmatrix} \boldsymbol{H}_a \\ \boldsymbol{I}_\kappa \\ -\boldsymbol{I}_\kappa \end{pmatrix},\ \boldsymbol{k}_p = \begin{pmatrix} \boldsymbol{k}_a \\ (\bar{\boldsymbol{r}}_z + \delta) \\ (-\bar{\boldsymbol{r}}_z + \delta) \end{pmatrix}.$$

The scalar $\delta$ is a small positive real number added element-wise to $\bar{\boldsymbol{r}}_z$ in order to avoid zero volume polytopes, which cause numerical problems. The resulting polytope is

$$\mathcal{Z}_p = \{z \in \mathbb{R}^p : \boldsymbol{H}_p z \le \boldsymbol{k}_p\}. \tag{7.17}$$

The feasible setpoint equilibrium $\bar{\boldsymbol{r}}_a$ is obtained from

$$\bar{\boldsymbol{r}}_a = \arg\min_{z\in\mathcal{Z}_p} \|(z - \bar{\boldsymbol{r}})\|. \tag{7.18}$$

The setpoint projection procedure is summarised in Algorithm 7.1 and its embedding into the reconfigurable control scheme is illustrated in Fig. 7.3.

Next, it is shown how a minimum equilibrium-preserving input matrix can be generated from a general input matrix. The guideline in preferring certain inputs over others consists in letting the virtual actuator prefer those actuators whose equilibrium values are centered between their actuation range limits. For a perfectly centered input $u_i$, the relation $\underline{u}_i + \overline{u}_i = 0$ holds. This goal is achieved by iteratively zeroing columns of $\boldsymbol{B}_f$ that correspond to non-centered actuators, while preserving the controllability and satisfying Condition (4.54). The online procedure is provided in Algorithm 7.2.

In Algorithm 7.2, the notation $\hat{\boldsymbol{B}}_f(m) := \boldsymbol{0}$ denotes assignment of the null-vector to the $m$-th column of $\hat{\boldsymbol{B}}_f$. However, it remains to mention that the application of Algorithm 7.2 in general reduces the feasible set $\mathcal{Z}_a$.

The overall synthesis and online application of the saturated virtual actuator is summarised in Algorithm 7.3.

**Example 7.1 (Tracking saturated virtual actuator synthesis for the ship).** *Given the elements (4.8), (4.9) of the linear model, Algorithm 7.3 is applied to the rudder failure $f_3$ combined with reduced actuation range of the left thruster $f_4$.*

*The application of the tracking recovery algorithm requires some care. Recall that the plant is assumed to have finite static gain from the input $\boldsymbol{u}$ to the controlled*

**Algorithm 7.1.** Online setpoint projection to admissible set

**Require:** Plant model $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{B}_f, \boldsymbol{C}_z)$, saturation bounds $\underline{\boldsymbol{u}}$, $\overline{\boldsymbol{u}}$, $\underline{\boldsymbol{u}}_f$, $\overline{\boldsymbol{u}}_f$, setpoint $\bar{\boldsymbol{r}}$, parameters $\boldsymbol{M}$, $\boldsymbol{N}$ of virtual actuator, set of generated inputs $\mathcal{U}_a$, set of safe outputs $\mathcal{Z}_s$

1: Determine $\mathcal{U}$ from Eq. (7.10) and determine $\kappa$ from Eq. (7.15)
2: Compute $\mathcal{Z}_a$ from $\mathcal{Z}_s$ and Eqs. (7.12), (7.13), (7.14)
3: **if** $\bar{\boldsymbol{r}} \in \mathcal{Z}_a$ **then**
4: **return** $\bar{\boldsymbol{r}}_a = \bar{\boldsymbol{r}}$
5: **end if**
6: Select $\kappa$ relevant outputs and determine $\boldsymbol{T}_z$ defined in Eq. (7.16)
7: **if** no priorities are assigned to any outputs **then**
8: Project $\bar{\boldsymbol{r}}$ onto $\mathcal{Z}_p$ by Eq. (7.18) with $\mathcal{Z}_p = \mathcal{Z}_a$
9: **else**
10: Project $\bar{\boldsymbol{r}}$ onto $\mathcal{Z}_p$ by Eq. (7.18) with $\mathcal{Z}_p$ as in Eq. (7.17)
11: **end if**

**Result:** Feasible setpoint $\bar{\boldsymbol{r}}_a$.

**Algorithm 7.2.** Online input matrix reduction

**Require:** Faulty plant model $(\boldsymbol{A}, \boldsymbol{B}_f)$, saturation bounds $\underline{\boldsymbol{u}}_f$, $\overline{\boldsymbol{u}}_f$

1: Let $\hat{\boldsymbol{u}} = \overline{\boldsymbol{u}}_f + \underline{\boldsymbol{u}}_f$
2: Compute candidate inputs for deletion,

$$m = \arg\max_j |\hat{u}_j|; \quad \hat{\boldsymbol{B}}_f := \boldsymbol{B}_f; \quad \hat{\boldsymbol{B}}_f(m) := \boldsymbol{0}.$$

3: **if** Eq. (4.54) is satisfied with $\hat{\boldsymbol{B}}_f$ instead of $\boldsymbol{B}_f$ **then**
4: Set $\boldsymbol{B}_f := \hat{\boldsymbol{B}}_f$ and goto step 2; **else** end
5: **end if**

**Result:** Reduced input matrix $\boldsymbol{B}_f$ that permits the same reconfiguration result w.r.t. stability and equilibrium recovery as the original input matrix.

*output $\boldsymbol{z}$ (Assumption 7.2). This assumption is violated if the heading is included in the controlled output vector. Therefore, the output $\boldsymbol{z}$ is chosen as $\boldsymbol{z} = (v\ r)^T$ for the purpose of this algorithm. In other words, the recovery of the yaw velocity is sought instead of the recovery of heading. This step is not restrictive, since a cascaded controller is used as a nominal controller that performs heading control in the outer loop, whereas an inner loop regulates the yaw velocity.*

*The response of the nonlinear ship model subject to abrupt rudder failure at $t_f = 20$ s is shown in Fig. 7.4. The times $0 \le t < t_f = 20$ s correspond to the steps 1 to 4 of the algorithm describing the situation without faults, where the virtual sensor is inactive. At the time $t_f = 20$ s, the rudder fails in floating condition. The fault is immediately detected and isolated in step 4, the model of the faulty system is constructed in step 5, and the virtual actuator parameters $\boldsymbol{M}$ and $\boldsymbol{N}$ are calculated*

**Algorithm 7.3.** Tracking-recovering saturated virtual actuator

**Require:** Plant model $(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$, saturation bounds $\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}$, initial condition $\boldsymbol{x}_0$

1: Initialise the nominal closed-loop system (5.4), (7.1), (7.5) with $\boldsymbol{B}_f = \boldsymbol{B}$, $\underline{\boldsymbol{u}}_f = \underline{\boldsymbol{u}}$, $\overline{\boldsymbol{u}}_f = \overline{\boldsymbol{u}}$, $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{N} = \boldsymbol{I}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, $\boldsymbol{x}_\Delta(t_0) = \boldsymbol{0}$
2: **repeat**
3: Run the nominal closed-loop system
4: **until** actuator fault $f$ detected and isolated
5: Construct the actuator fault model $\boldsymbol{B}_f$, $\underline{\boldsymbol{u}}_f$, $\overline{\boldsymbol{u}}_f$, the sector bound $\boldsymbol{S}_{\boldsymbol{\varphi}} = \boldsymbol{I}$, and update the saturated virtual actuator (7.5)
6: Compute feasible solutions $(\boldsymbol{X}_a, \boldsymbol{Y}_a)$ of the linear matrix inequality (6.11)
7: Compute reduced faulty input matrix $\hat{\boldsymbol{B}}_f$ using Alg. 7.2
8: Compute the gain $\boldsymbol{N}$ from Eq. (7.8) where $\boldsymbol{B}_f$ is replaced by $\hat{\boldsymbol{B}}_f$
9: Update the saturated virtual actuator with $\boldsymbol{M} = \boldsymbol{Y}_a \boldsymbol{X}_a^{-1}$, $\boldsymbol{N}$
10: Compute feasible setpoint $\bar{\boldsymbol{r}}_a$ from current setpoint $\bar{\boldsymbol{r}}$ using Alg. 7.1
11: **repeat**
12: Run the reconfigured closed-loop system
13: **if** the setpoint $\bar{\boldsymbol{r}}$ changes **then**
14: Recompute feasible setpoint $\bar{\boldsymbol{r}}_a$ from current setpoint $\bar{\boldsymbol{r}}$ using Alg. 7.1
15: **end if**
16: **until** closed-loop system stopped

**Result:** Globally ISS and setpoint-tracking reconfigured closed-loop system (5.4), (7.1), (7.5)

*in steps 6 to 9, where the result for $\boldsymbol{M}$ is the same that was given in Equation (6.15), and the feedthrough gain is*

$$\boldsymbol{N} = \begin{pmatrix} 1 & 0 & 5 \\ 0 & 1 & -5 \\ 0 & 0 & 0 \end{pmatrix}. \tag{7.19}$$

*All setpoints for heading and velocity are still feasible, so step 10 does not change the reference signal. The reconfigured closed-loop system is executed in step 7.*

*The matrix $\boldsymbol{N}$ can be interpreted as follows. The thrust inputs from the nominal controller (first two columns) are fed through, which is logical, since these inputs determine the surge velocity, and since the thrusters are not faulty. The last column means that a positive rudder angle (steer to the right) is translated into a left thruster force increase and a right thruster force decrease, which also cases positive momentum and steers the ship to the right.*

*Fig. 7.4 shows that, indeed, all variables are stable, and the reference input (dashed) for the surge velocity (solid) is asymptotically recovered. The sway and yaw velocities quickly converge to the origin. The heading is not shown in that figure but indicated in the following figure.*

*Fig. 7.5 shows the motion of the ship with reconfigured controller, which is capable of avoiding the obstacle. However, it does not return to its old course after*

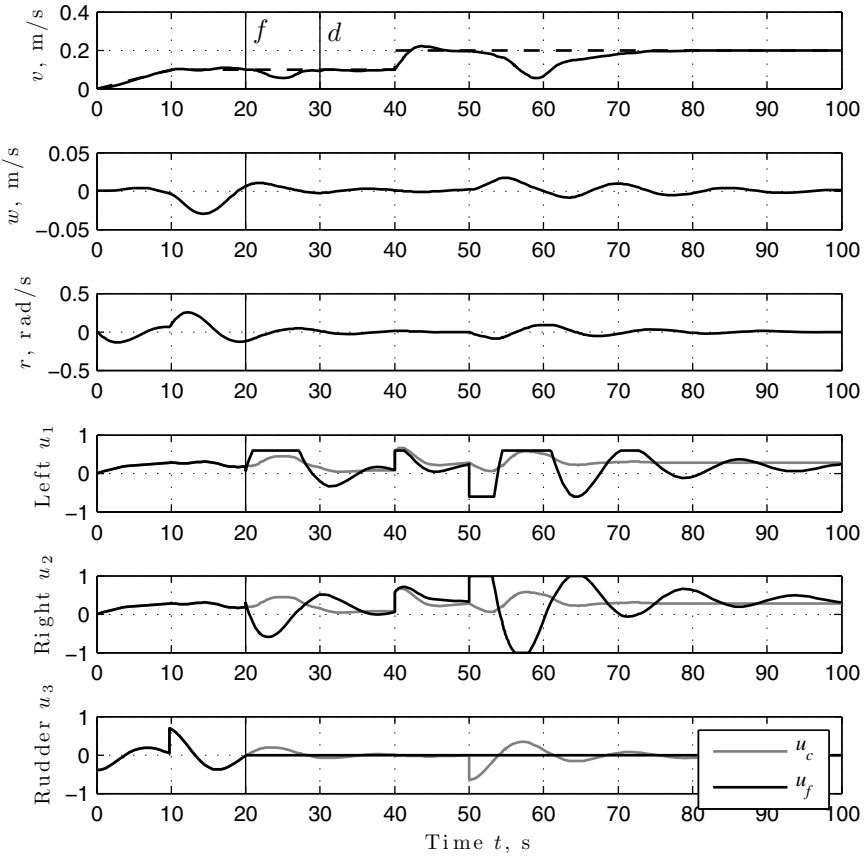


**Fig. 7.4** Response of nonlinear ship subject to rudder failure and control reconfiguration by means of tracking-recovering saturated virtual actuator.

*avoiding the obstacle, therefore the solution is not yet satisfactory. This shortcoming is due to the small gain of the feedback gain* $\boldsymbol{M}$*, which results in a too slow response of the virtual actuator.*

## 7.5 Summary and Discussion

In this chapter, it was shown how the linear solution to the tracking recovery problem can be extended to solve the corresponding Problem 5.2 in saturated systems (Theorem 7.1). The main idea consists in the expression of saturation operators by intersection operations. The approach requires few additional steps, limited to the online projection of infeasible setpoints to feasible setpoints (Algorithm 7.1), and the online reduction of the input matrix of the faulty saturated system (Algorithm 7.2).

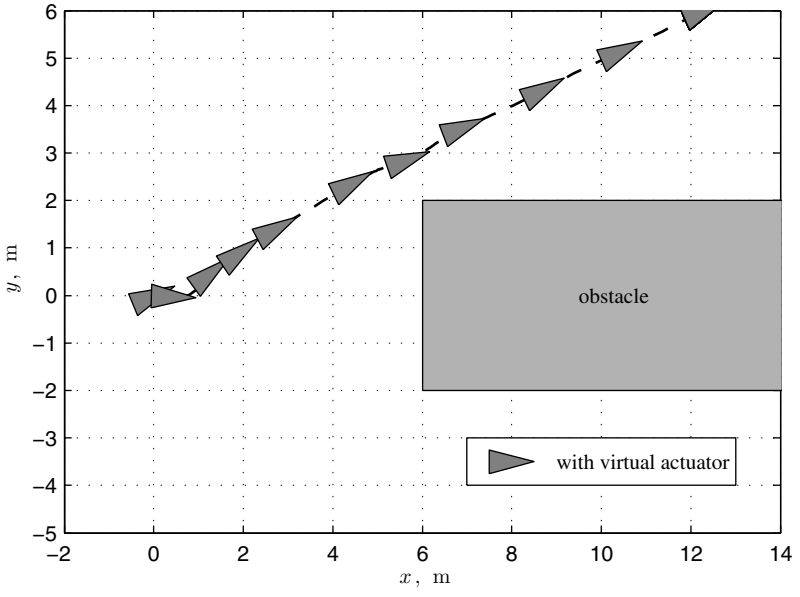


**Fig. 7.5** Motion of nonlinear ship subject to rudder failure and control reconfiguration by means of tracking-recovering Hammerstein virtual actuator.

Due to the input matrix reduction, some conservatism is introduced. In other words, in general, for any input matrix reduction there exist equilibria that cannot be reached with the reduced input matrix, but could be reached with a differently reduced input matrix. In order to be able to reach every setpoint that is feasible with the un-reduced input matrix, the feedforward gain $\boldsymbol{N}$ of the virtual actuator would have to depend on the setpoint. The idea of making the feedforward gain $\boldsymbol{N}$ dependent on the reference input represents a potential extension of the presented approach. Care must be taken to ensure that perpetual switching between different values for $\boldsymbol{N}$ does not cause unstable behaviour.

The following chapter describes the recovery of nominal performance. The motivation for considering performance recovery is highlighted by Example 7.1.

# Chapter 8
# Performance Recovery after Actuator Faults in Saturated Systems

**Abstract.** This chapter gives the solution to the optimal performance recovery problem after actuator faults in saturated systems. The solution extends the stabilising reconfiguration solution to achieve a compromise between output trajectory recovery and control input amplification.

## 8.1 Overview of the Performance Recovery Problem and Its Solution

The task considered in this chapter consists in solving Problem 5.3, which requires reaching an optimal compromise between small input amplification and good performance recovery, while preserving the reconfigured closed-loop ISS condition established in Corollary 6.1. The class of systems considered in this chapter is the same as that considered in the previous Chapter 7, namely nominal saturated systems (5.3) and saturated systems subject to actuator faults (5.9).

It was found that if the virtual actuator gain $\boldsymbol{M} = \boldsymbol{Y}_a \boldsymbol{X}_a^{-1}$ is chosen such that the variables $\boldsymbol{X}_a, \boldsymbol{Y}_a$ satisfy the LMI (6.11), stated here specifically for saturated systems where $\boldsymbol{S}_\varphi = \boldsymbol{I}_m$

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T) & \boldsymbol{Y}_a^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array}\right) \succ 0, \tag{8.1}$$

then the reconfigured closed-loop ISS is guaranteed. The number of feasible solutions to the LMI (8.1) is infinite, which leaves considerable freedom for good and bad choices of $\boldsymbol{M}$ with respect to Problem 5.3.

The approach to attaining a good compromise between performance and control effort, as required by Problem 5.3, consists in the consideration of the linear subsystems of the saturated plant and the saturated virtual actuator. Linear optimisation approaches are used to minimise the $H_\infty$-norms of transfer functions that correspond to the signal paths describing the problem. All statements about optimality made throughout this chapter are valid only if the control inputs do not exceed the

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 131–140.
springerlink.com 

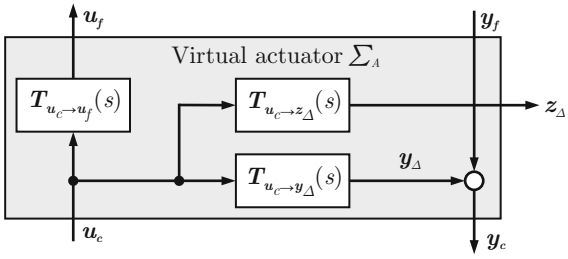


**Fig. 8.1** Transfer functions describing the linear virtual actuator.

saturation bounds imposed on the inputs. Condition (8.1), however, guarantees that stability is preserved also if saturation bounds are exceeded.

The considered linear subsystems are the nominal linear plant (4.1), the faulty linear plant (4.14), and the linear virtual actuator (4.49), summarised here again for ease of reference,

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B_d}\boldsymbol{d}(t),\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t) \end{cases}$$

$$\Sigma_{Pf} : \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{B_d}\boldsymbol{d}(t),\ \boldsymbol{x}_f(0) = \boldsymbol{x}_0 \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t) \end{cases}$$

$$\Sigma_A : \begin{cases} \dot{\boldsymbol{x}}_\Delta(t) &= \boldsymbol{A}_\Delta\boldsymbol{x}_\Delta(t) + \boldsymbol{B}_\Delta\boldsymbol{u}_c(t),\ \boldsymbol{x}_\Delta(0) = \boldsymbol{0} \\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{N}\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) &= \boldsymbol{C}\boldsymbol{x}_\Delta(t) + \boldsymbol{y}_f(t). \end{cases}$$

The output performance recovery is characterised by the transfer function

$$\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_\Delta}(s) = \boldsymbol{C}_z\left(s\boldsymbol{I} - (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})\right)^{-1}(\boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}) \tag{8.2}$$

of the virtual actuator (Fig. 8.1). The input amplification by the virtual actuator is characterised by the transfer function

$$\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(s) = \boldsymbol{M}\left(s\boldsymbol{I} - (\boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M})\right)^{-1}(\boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}) + \boldsymbol{N} \tag{8.3}$$

of the virtual actuator (Fig. 8.1).

With these transfer functions, it is possible to re-state and solve the parts of Problem 5.3, which consist in limiting the gains of the stated transfer functions in the sense of their $H_\infty$-norms. This norm was chosen because it represents the largest peak-to-peak amplification of harmonic input signals by the system over the entire frequency range. The $H_2$-norm is an alternative criterion that represents the

asymptotic output variance of the transfer function driven by white noise. Since the control input signal $\boldsymbol{u}_c$ is not stochastic, the $H_\infty$-norm is chosen here.

## 8.2 Output Trajectory Recovery

This section focusses entirely on the output trajectory recovery part of Problem 5.3. The goal is to find suitable gains $\boldsymbol{M}$ and $\boldsymbol{N}$ such that the $H_\infty$-norm $\gamma_z$ of the transfer function (8.2) is minimised, while preserving the closed-loop ISS property of the system with saturations. In order to give priority to certain components of $\boldsymbol{z}$ over others, a symmetric weight matrix $\boldsymbol{Q} = \boldsymbol{Q}^T > 0$ is left-multiplied to the transfer function $\boldsymbol{T}_{\boldsymbol{u}_c \to z_\Delta}(s)$ defined in Equation (8.2). For that goal, Problem 5.3 simplifies to the following problem.

**Problem 8.1 (Optimal output trajectory recovery).** Given a weight $\boldsymbol{Q} = \boldsymbol{Q}^T > 0$, find virtual actuator gains $\boldsymbol{M}$ and $\boldsymbol{N}$ that solve the optimisation problem

$$\min_{\boldsymbol{N}, \boldsymbol{X}_a, \boldsymbol{Y}_a} \left\| \boldsymbol{Q} \boldsymbol{T}_{\boldsymbol{u}_c \to z_\Delta}(s) \right\|_{H_\infty} = \gamma_z$$

subject to

$$\left( \begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a \boldsymbol{A}^T) & \boldsymbol{Y}_a^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array} \right) > 0,$$

where $\boldsymbol{X}_a = \boldsymbol{X}_a^T > 0$, and $\boldsymbol{M} = \boldsymbol{Y}_a \boldsymbol{X}_a^{-1}$.

The solution to Problem 8.1 is given by the following theorem, which also solves Problem 5.3 specified for pure output trajectory recovery.

**Theorem 8.1 (Optimal output recovering saturated virtual actuator synthesis).** *The reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_A, \Sigma_C)$ that consists of the saturated system (5.3), the saturated virtual actuator (7.5), and the nominal controller (5.4) optimally recovers the nominal output performance in the sense of the $H_\infty$-norm of the corresponding weighed transfer function (8.2), if the matrices $\boldsymbol{M}$ and $\boldsymbol{N}$ of the virtual actuator are solutions of the optimisation problem*

$$\min_{\boldsymbol{N}, \boldsymbol{X}_{az}, \boldsymbol{Y}_{az}} \gamma_z$$

*subject to*

$$\left(\begin{array}{c|c|c} \boldsymbol{A}\boldsymbol{X}_{az} + \boldsymbol{X}_{az}\boldsymbol{A}^T - \boldsymbol{B}_f\boldsymbol{Y}_{az} - \boldsymbol{Y}_{az}^T\boldsymbol{B}_f^T & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{X}_{az}\boldsymbol{C}_z^T\boldsymbol{Q} \\ \hline \star & -\gamma_z\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_z\boldsymbol{I} \end{array}\right) \prec 0 \qquad (8.4)$$

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_{az} + \boldsymbol{X}_{az}\boldsymbol{A}^T) & \boldsymbol{Y}_{az}^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array}\right) \succ 0$$

$$\boldsymbol{X}_{az} = \boldsymbol{X}_{az}^T \succ 0,\ \gamma_z > 0,$$

*where* $\boldsymbol{M} = \boldsymbol{Y}_{az}\boldsymbol{X}_{az}^{-1}$*, and if the control inputs* $\boldsymbol{u}_c$ *and* $\boldsymbol{u}_f$ *do not exceed the saturation bounds. Furthermore, the reconfigured closed-loop system is ISS w.r.t. its input* $(\boldsymbol{r}, \boldsymbol{d})$*.*

*Proof.* See Appendix D, page 264.

The optimisation problem to be solved in Theorem 8.1 has a linear objective function and is defined on a convex feasible region. This class of problems is readily solvable using numerical methods.

The exclusive focus on performance generally results in high-amplitude input signals, which are not realisable due to saturations. The next section discusses the miminisation of the input energy amplification by the virtual actuator.

## 8.3 Input Energy Limitation

In this section, the opposite special case of the previous section is solved, namely minimising the input amplification of the virtual actuator, as formulated as part of Problem 5.3. The goal is to find suitable gains $\boldsymbol{M}$ and $\boldsymbol{N}$ such that the $H_\infty$-norm $\gamma_u$ of the transfer function (8.3) is minimised, while preserving the closed-loop ISS property of the system with saturations. For this goal, Problem 5.3 simplifies to the following problem.

**Problem 8.2 (Minimum input amplification).** Find virtual actuator gains $\boldsymbol{M}$ and $\boldsymbol{N}$ that solve the optimisation problem

$$\min_{\boldsymbol{N},\boldsymbol{X}_a,\boldsymbol{Y}_a} \left\| \boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(s) \right\|_{H_\infty} = \gamma_u$$

subject to

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T) & \boldsymbol{Y}^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array}\right) \succ 0,$$

where $\boldsymbol{X}_a = \boldsymbol{X}_a^T \succ 0$, and $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$.

**Theorem 8.2 (Minimum input amplification virtual actuator).** *The reconfigured closed-loop system* $(\Sigma_{Pf}, \Sigma_A, \Sigma_C)$ *that consists of the saturated system (5.3), the saturated virtual actuator (7.5), and the nominal controller (5.4) minimally amplifies the control input in the sense of the* $H_\infty$*-norm of the corresponding transfer function (8.3), if the matrices* $\boldsymbol{M}$ *and* $\boldsymbol{N}$ *of the virtual actuator are solutions of the optimisation problem*

$$\min_{\boldsymbol{N}, \boldsymbol{X}_{au}, \boldsymbol{Y}_{au}} \gamma_u$$

*subject to*

$$\left(\begin{array}{c|c|c} \boldsymbol{A}\boldsymbol{X}_{au} + \boldsymbol{X}_{au}\boldsymbol{A}^T - \boldsymbol{B}_f\boldsymbol{Y}_{au} - \boldsymbol{Y}_{au}^T\boldsymbol{B}_f^T & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{Y}_{au}^T \\ \hline \star & -\gamma_u\boldsymbol{I} & \boldsymbol{N}^T \\ \hline \star & \star & -\gamma_u\boldsymbol{I} \end{array}\right) \prec 0 \qquad (8.5)$$

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_{au} + \boldsymbol{X}_{au}\boldsymbol{A}^T) & \boldsymbol{Y}_{au}^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array}\right) \succ 0$$

$$\boldsymbol{X}_{au} = \boldsymbol{X}_{au}^T \succ 0,\ \gamma_u \geq 1,$$

*where* $\boldsymbol{M} = \boldsymbol{Y}_{au}\boldsymbol{X}_{au}^{-1}$*, and if the control inputs* $\boldsymbol{u}_c$ *and* $\boldsymbol{u}_f$ *do not exceed the saturation bounds. Furthermore, the reconfigured closed-loop system is ISS w.r.t. its input* $(\boldsymbol{r}, \boldsymbol{d})$*.*

*Proof.* See Appendix D, page 264.

Again, the optimisation problem to be solved in Theorem 8.2 has a linear objective function and is defined on a convex feasible region. The result allows the minimisation of input energy amplification by the virtual actuator. The constraint $\gamma_u \geq 1$ ensures that the optimal solution obtained from Theorem 8.2 is not $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{N} = \boldsymbol{0}$. This solution is not desirable in practice, since it disconnects the faulty saturated plant from the controller and lets the faulty plant run in open loop.

The exclusive minimisation of input signal amplification might lead to insufficient performance. A practical solution will, therefore, have to combine the ideas shown in Section 8.2 and the present section. That approach is shown in the next section.

## 8.4 Weighed Multi-objective Synthesis

Neither the exclusive consideration of performance nor the exclusive consideration of input energy are useful for practical design of the virtual actuator. This section combines both objectives by means of weighing into a single synthesis objective that

reflects a compromise between a small input amplification and a small output error, thus providing a complete solution to Problem 5.3. The weight $\lambda$ is a free parameter specified by the user and can be adjusted to the needs of the considered application. In particular, the suitable choice of the weight parameter $\lambda$ helps satisfying the assumption that the control inputs do not exceed the saturation bounds.

**Theorem 8.3 (Weighed multi-objective virtual actuator synthesis).** *Consider the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_A, \Sigma_C)$ that consists of the saturated system (5.3), the saturated virtual actuator (7.5), and the nominal controller (5.4). The reconfiguration solution realises an optimal compromise $\lambda\gamma_z + (1-\lambda)\gamma_u$ between output recovery and input amplification expressed by means of the weight parameter $\lambda \in [0,1]$, if the degrees of freedom $\boldsymbol{M}$ and $\boldsymbol{N}$ are solutions of the optimisation problem*

$$\min_{\boldsymbol{N},\boldsymbol{X}_a,\boldsymbol{Y}_a} \lambda\gamma_z + (1-\lambda)\gamma_u \tag{8.6}$$

*subject to*

$$\left(\begin{array}{c|c|c} \boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T - \boldsymbol{B}_f\boldsymbol{Y}_a - \boldsymbol{Y}_a^T\boldsymbol{B}_f^T & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{Y}_a^T \\ \hline \star & -\gamma_u\boldsymbol{I} & \boldsymbol{N}^T \\ \hline \star & \star & -\gamma_u\boldsymbol{I} \end{array}\right) < 0 \tag{8.7}$$

$$\left(\begin{array}{c|c|c} \boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T - \boldsymbol{B}_f\boldsymbol{Y}_a - \boldsymbol{Y}_a^T\boldsymbol{B}_f^T & \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N} & \boldsymbol{X}_a\boldsymbol{C}_z^T\boldsymbol{Q} \\ \hline \star & -\gamma_z\boldsymbol{I} & \boldsymbol{0} \\ \hline \star & \star & -\gamma_z\boldsymbol{I} \end{array}\right) < 0 \tag{8.8}$$

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}^T) & \boldsymbol{Y}_a^T - \boldsymbol{B}_f \\ \hline \star & 2\boldsymbol{I}_m \end{array}\right) > 0 \tag{8.9}$$

$$\boldsymbol{X}_a = \boldsymbol{X}_a^T > 0,\ \gamma_u \geq 1,\ \gamma_z > 0, \tag{8.10}$$

*where $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$. The weight parameter $\lambda$ is specified by the user. Optimality is achieved if the inputs $\boldsymbol{u}_c$ and $\boldsymbol{u}_f$ do not exceed the saturation bounds. Furthermore, the reconfigured closed-loop system is ISS w.r.t. its input $(\boldsymbol{r}, \boldsymbol{d})$.*

*Proof.* See Appendix D, page 265.

The case $\lambda = 0$ corresponds to minimum actuation energy amplification, whereas the case $\lambda = 1$ corresponds to minimum performance loss. Like the separate single-objective approaches, the underlying optimisation problem in this multi-objective approach is linear and convex. The optimisation problem embedded in the previous theorem gives rise to Pareto-optimal solutions, meaning that an improvement of one of the included objectives implies a degradation of the other objective. In practice, it is advantageous to choose the weight $\lambda$ small enough so that the control inputs do not exceed the saturation bounds under typical operating conditions.

**Algorithm 8.1.** Performance-recovering saturated virtual actuator with limited input amplification

**Require:** Plant model $(\boldsymbol{A},\boldsymbol{B},\boldsymbol{C})$, saturation bounds $\underline{\boldsymbol{u}}_f$, $\overline{\boldsymbol{u}}_f$, performance/energy weight $\lambda$, output weight matrix $\boldsymbol{Q}$, initial condition $\boldsymbol{x}_0$

1: Initialise the nominal closed-loop system (5.4), (7.1), (7.5) with $\boldsymbol{B}_f = \boldsymbol{B}$, $\underline{\boldsymbol{u}}_f = \underline{\boldsymbol{u}}$, $\overline{\boldsymbol{u}}_f = \overline{\boldsymbol{u}}$, $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{N} = \boldsymbol{I}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, $\boldsymbol{x}_\Delta(t_0) = \boldsymbol{0}$
2: **repeat**
3: Run the nominal closed-loop system
4: **until** actuator fault $f$ detected and isolated
5: Construct the actuator fault model $\boldsymbol{B}_f$, $\underline{\boldsymbol{u}}_f$, $\overline{\boldsymbol{u}}_f$ and update the saturated virtual actuator (7.5)
6: Solve the optimisation problem (8.7) subject to the semidefinite constraints (8.7)–(8.10) for the variables $\boldsymbol{X}_a, \boldsymbol{Y}_a, \boldsymbol{N}$
7: Update the saturated virtual actuator with $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$ and $\boldsymbol{N}$
8: Run the reconfigured closed-loop system

**Result:** Globally ISS reconfigured closed-loop system (5.4), (7.1), (7.5) that optimally realises the chosen compromise between performance recovery and minimum input amplification

The LMI-based characterisations of the $H_\infty$-norms of the transfer functions $\boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{z}_\Delta}(s)$ and $\boldsymbol{T}_{\boldsymbol{u}_c\to\boldsymbol{u}_f}(s)$ use different variables $\boldsymbol{X}_{az}$ and $\boldsymbol{X}_{au}$ as well as $\boldsymbol{Y}_{az}$ and $\boldsymbol{Y}_{au}$, respectively. The multiobjective optimisation problem (8.7) could only be formulated with consistent constraints after unification of variables $\boldsymbol{X}_a = \boldsymbol{X}_{az} = \boldsymbol{X}_{au}$ and $\boldsymbol{Y}_a = \boldsymbol{Y}_{az} = \boldsymbol{Y}_{au}$ (see the proof of Theorem 8.3). This simplification introduces some conservatism. In other words, the computed Pareto-optimal solutions might not be globally optimal. However, the true global optimum cannot be computed, and the chosen approach provides a good approximation. Note also that not the optimal gains themselves, but the corresponding solution variables are of main interest here.

The overall online application of the weighed multi-objective synthesis of the saturated virtual actuator for saturated systems is summarised in Algorithm 8.1.

**Example 8.1 (Performance recovering saturated virtual actuator synthesis for the ship).** *In Example 7.1, it became obvious that stability and tracking are not always sufficiently adequate in practice. Therefore, performance recovery is now applied to the ship after rudder failure* $f_3$.

*The response of the nonlinear ship model subject to abrupt rudder failure at* $t_f = 20$ *s is shown in Fig. 8.2. Before the fault occurs at* $t_f = 20$ *s, the saturated virtual actuator is inactive due to the choice of its parameters* $\boldsymbol{M} = \boldsymbol{0}$ *and* $\boldsymbol{N} = \boldsymbol{I}$. *The time period* $0 \le t < t_f$ *corresponds to the steps 1 to 4 of Algorithm 8.1. The fault that occurs at* $t_f = 20$ *s is assumed to be immediately detected and isolated in step 5 of the algorithm. Given the elements (4.8), (4.9) of the linear model, Algorithm 8.1 is applied with the tradeoff weight* $\lambda = 0.2$, *thus with focus on limiting the control input amplification, and with the output weight* $\boldsymbol{Q} = \mathrm{diag}(1\ 0.1)$. *In steps 6 and 7, the algorithm computes the feedback gain* $\boldsymbol{M}$ *and the feedforward gain* $\boldsymbol{N}$

$$\boldsymbol{M} = \begin{pmatrix} 1.59 & -331.6 & 244.1 & 40.2 \\ 1.59 & 331.6 & -244.1 & -40.2 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \; \boldsymbol{N} = \begin{pmatrix} 0.94 & 0.038 & 4.53 \\ 0.038 & 0.94 & -4.53 \\ 4.53 & -4.53 & -0.91 \end{pmatrix}$$

*and updates the saturated virtual actuator. The ship is governed by reconfigured closed-loop dynamics in step 8, which covers the time interval $t_f = 20\ s \leq t < \infty$.*

*It is clearly visible from Fig. 8.2 that the thrusters replace the failed rudder in order to manipulate the ship yaw velocity and heading. All signals are stable, the surge velocity follows its reference input, and the ship follows its path while avoiding the obstacle, as Fig. 8.3 clearly shows. This example highlights the role of performance recovery in the context of reconfigurable control.*

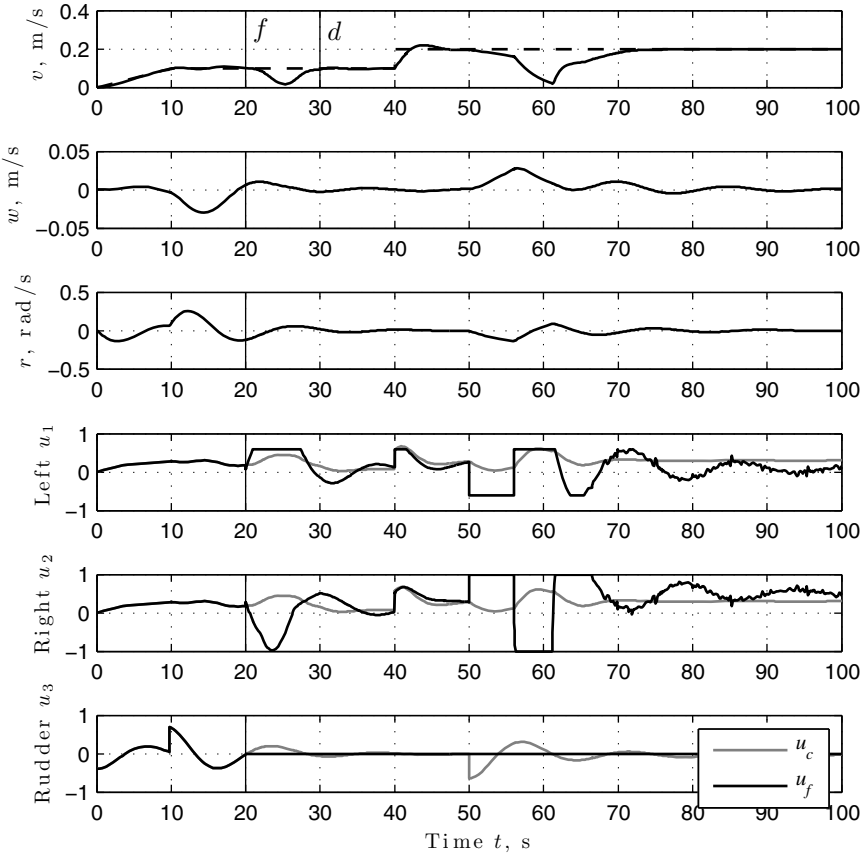


**Fig. 8.2** Response of nonlinear ship subject to rudder failure with control reconfiguration by means of performance-recovering saturated virtual actuator.

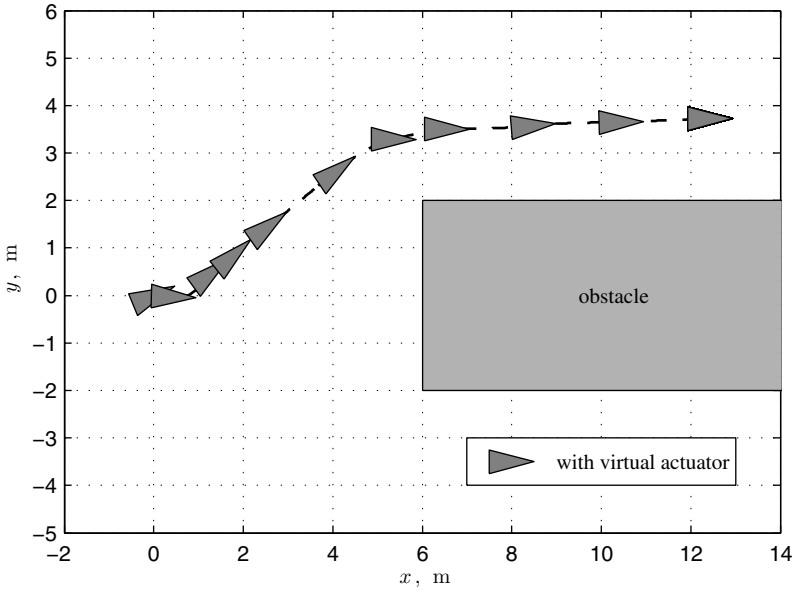


**Fig. 8.3** Motion of nonlinear ship subject to rudder failure with control reconfiguration by means of performance-recovering saturated virtual actuator.

## 8.5 Summary and Discussion

In this chapter, it was shown how the linear solution to the performance recovery problem can be extended to solve the corresponding Problem 5.3 in saturated systems subject to actuator faults. The main idea consists in the consideration of the linear dynamics of the saturated system, and a compromise between performance recovery (Theorem 8.1) and input signal amplification (Theorem 8.2) expressed in Theorem 8.3. If the nominal controller respects the saturations, then a cautious adjustment of the weight $\lambda$ in the reconfiguration Algorithm 8.1 can prevent excessive input signals. The inclusion of the stability condition (8.1) for the saturated system ensures that the reconfigured closed-loop system remains stable, also if the feasible inputs are exceeded. As mentioned above, optimality is attained only if the input signals do not exceed the saturation bounds.

With respect to the requirements stated in Section 1.3, the method is well-suited for autonomous online application. The only choices to be made by the user are the weight $\lambda$ between input amplification and output performance, and the diagonal output weight matrix $\boldsymbol{Q}$. Therefore, the choice between potentially large numbers and very different kinds of reconfigured controllers has been reduced to the choice of a scalar number that has an intuitive interpretation as a weight between contradicting goals, and a well-interpretable matrix placing emphasis on certain outputs over others. Given the current computational power, the approach is numerically

applicable to small and medium sized problems, and the robustness properties derived in Chapter 6 remain valid.

A possible extension of the performance recovering saturated virtual actuator consists in the inclusion of integral action by means of extension of the virtual actuator state. The resulting transfer functions that define the optimisation problem still define systems with well-defined $H_\infty$-norms. The extended system matrix is, however, no longer Hurwitz, and a slightly different technical statement about the closed-loop stability must be made, in the sense that only convergence of the extended state to a neighborhood of the origin can be verified. This result is logical and to be expected from the inclusion of pure integrators into the difference system. However, the mentioned extension is often useful in practice.

This chapter concludes Part II of this monograph on the reconfigurable control of Hammerstein-Wiener systems. The solutions presented in this part are linked as follows. In many practical applications, the most important limitation to the validity of linear models is the presence of input saturations. In this case, saturated systems adequately describe the system, and the stable setpoint tracking and optimal performance recovering reconfiguration solutions of Chapters 7 and 8 can be applied. The robustness properties found in Chapter 6 furthermore provide safeguard against model errors that result from both inaccurate diagnosis results and from neglected moderate (static) nonlinear effects.

In the case of additional sensor faults, a saturated virtual sensor can be easily derived as a special case of the Hammerstein-Wiener virtual sensor. The results on setpoint tracking and optimal performance recovery remain valid when coupling the virtual sensor with the virtual actuator in the absence of disturbances. The presence of disturbances adds difficulty to the setpoint tracking recovery, since disturbances force the observation error and consequently also the tracking error away from the origin. Unknown-input (PI) observers for saturated systems would have to be used to suppress the disturbance effect on the tracking error.

The class of Hammerstein-Wiener systems studied in this part does not reflect nonlinear dynamics. If nonlinear dynamics play a major role in the considered system, or if large operation ranges have to be considered, then the methods presented in this part have limited utility. The following Part III of this monograph thus extends the fault-hiding approach to reconfigurable control to piecewise affine systems, which are considered as approximations of systems with input-affine nonlinear dynamics.

# Part III
# Reconfigurable Control of Piecewise Affine Systems

In this part, the reconfigurable control problem is solved for the class of piecewise affine systems, where the fault-hiding principle provides the conceptual basis. Actuator and sensor faults may occur simultaneously in this approach. Solutions are given with respect to the problems of recovering closed-loop stability and asymptotic setpoint tracking. The practically important case where a piecewise affine reconfiguration block is used in connection with a nonlinear plant is discussed.

# Chapter 9
# Control Reconfiguration Problem for Piecewise Affine Systems

**Abstract.** This chapter defines nominal piecewise affine systems and the nominal closed-loop system for the class of PWA systems, as well as the assumptions made about the nominal closed-loop system. It is shown how faults are modelled in piecewise affine systems, and the corresponding reconfiguration problems are formulated. The chapter closed with bibliographic notes.

## 9.1 Nominal Piecewise Affine Systems

A further extension of linear systems are piecewise affine systems, where the system matrix and an affine term are allowed to vary depending on the present state. The state-space is partitioned into a collection of $r$ non-overlapping polyhedra $\Lambda_i$ that cover the entire state-space, namely

$$\forall i \neq j : \ \operatorname{int}(\Lambda_i) \cap \operatorname{int}(\Lambda_j) = \emptyset$$
$$\bigcup_{i=1}^{r} \Lambda_i = \mathbb{R}^n.$$

**Definition 9.1 (Piecewise affine system [90, 197]).** A *piecewise affine (PWA) system* is a system of first-order ODEs

$$\Sigma_P : \begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{A}_i \boldsymbol{x}(t) + \boldsymbol{a}_i + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B}_d \boldsymbol{d}(t) \text{ for } \boldsymbol{x}(t) \in \Lambda_i,\ i \in \{1,\dots,r\} \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z \boldsymbol{x}(t) \end{cases} \tag{9.1}$$

where all signals are in accordance with Definition 3.2. The symbol $\boldsymbol{A}_i$, $i \in \{1,\dots,r\}$, denotes a family of system matrices, whereas $\boldsymbol{a}_i$, $i \in \{1,\dots,r\}$, is a family of affine terms. The remaining matrices are in accordance with Definition 4.1, all matrices are of compatible dimensions, and

$$\boldsymbol{x}(0) = \boldsymbol{x}_0$$

is the initial condition. ◇

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 143–155.
springerlink.com 

Each of the pairwise disjoint sets $\Lambda_k$ corresponds to a *mode* of the PWA system (9.1) in the sense that if $\boldsymbol{x}(t) \in \Lambda_k$ at time $t$, then the system is described by the $k$-th affine system represented by the tuple $(\boldsymbol{A}_k, \boldsymbol{a}_k, \boldsymbol{B}, \boldsymbol{B_d}, \boldsymbol{C}, \boldsymbol{C}_z)$ at time $t$. Switching among the modes is triggered when the state trajectory crosses a boundary between two polyhedra. A PWA system is shown in Fig. 9.1.

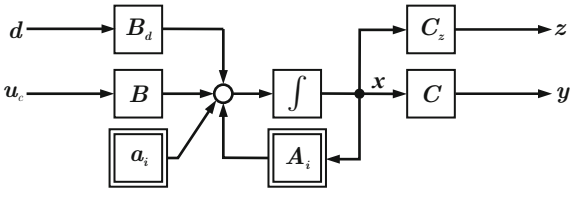


**Fig. 9.1** Piecewise affine system.

The following assumption is made throughout the monograph.

**Assumption 9.1.** *The right-hand side*

$$\boldsymbol{f}_a(\boldsymbol{x}, \boldsymbol{u}_c, \boldsymbol{d}) \triangleq \boldsymbol{A}_i \boldsymbol{x} + \boldsymbol{a}_i + \boldsymbol{B}\boldsymbol{u}_c + \boldsymbol{B_d}\boldsymbol{d} \text{ for } \boldsymbol{x} \in \Lambda_i,\ i \in \{1, \ldots, r\}$$

*of the system (9.1) is assumed to be a continuous function of its arguments* $\boldsymbol{x}$, $\boldsymbol{u}_c$ *and* $\boldsymbol{d}$.

The right-hand side of the PWA system (9.1) is automatically continuous for $\boldsymbol{x} \in \text{int}(\Lambda_i)$, $i \in \{1, \ldots, r\}$, since it is affine in every mode. Discontinuities may occur at the boundaries between adjacent polyhedra, and the system modeler must ensure that the PWA system model is free from such discontinuities. However, the right-hand side of the PWA system is nonsmooth on the boundaries between the mode-defining polyhedra. Note that Assumption 9.1 guarantees that the system (9.1) is locally Lipschitz-continuous. Consequently, for any $\boldsymbol{u}_c \in \mathcal{L}_1^{loc}(\mathbb{R}^m)$, $\boldsymbol{d} \in \mathcal{L}_1^{loc}(\mathbb{R}^k)$, and $\boldsymbol{x}_0 \in \mathbb{R}^n$, it has a unique and globally defined solution that is locally absolutely continuous. Also, sliding modes cannot occur as solutions of the PWA system (9.1) if it is continuous.

Note also that the original definition of PWA systems in [90, 197] allows switching input and output matrices. This extension, however, immediately destroys the continuity property, which is instrumental for obtaining the results of the following two chapters. Note that all papers that address the more general class of PWA systems provide solutions to the problem of stabilising equilibrium points. The reconfiguration problem, however, leads to a stabilisation problem of time-varying solutions of multi-mode PWA systems, for which only results based on continuity are known (solutions for bimodal discontinuous PWA systems are available, though).

The following theorem summarises prior results on the incremental stability, ISS, and convergence of continuous PWA systems.

**Theorem 9.1 (PWA incremental stability, ISS, and convergence [156, 159]).** *Consider the PWA system (9.1) with the right-hand side* $\boldsymbol{f}_a(\boldsymbol{x},\boldsymbol{u}_c,\boldsymbol{d}) \triangleq \boldsymbol{A}_i\boldsymbol{x}+\boldsymbol{a}_i+\boldsymbol{B}\boldsymbol{u}_c+\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}$ *for* $\boldsymbol{x}\in\Lambda_i$, $i\in\{1,\dots,r\}$, *and suppose that Assumption 9.1 holds. If there exists a matrix* $\boldsymbol{X}\in\mathbb{R}^{n\times n}$, $\boldsymbol{X}=\boldsymbol{X}^T\succ 0$ *that satisfies the LMIs*

$$\boldsymbol{X}\boldsymbol{A}_i+\boldsymbol{A}_i^T\boldsymbol{X}\prec 0,\ i=1,\dots,r, \tag{9.2}$$

*then the system (9.1) is 0-GES for* $\boldsymbol{u}_c,\boldsymbol{d}\equiv\boldsymbol{0}$, *globally ISS w.r.t.* $(\boldsymbol{u}_c,\boldsymbol{d})$, *and for any two points* $\boldsymbol{x}_1,\boldsymbol{x}_2\in\mathbb{R}^n$, *the following algebraic inequality holds:*

$$(\boldsymbol{x}_1-\boldsymbol{x}_2)^T\boldsymbol{X}(\boldsymbol{f}_a(\boldsymbol{x}_1,\boldsymbol{u}_c,\boldsymbol{d})-\boldsymbol{f}_a(\boldsymbol{x}_2,\boldsymbol{u}_c,\boldsymbol{d}))\le-\beta(\boldsymbol{x}_1-\boldsymbol{x}_2)^T\boldsymbol{X}(\boldsymbol{x}_1-\boldsymbol{x}_2). \tag{9.3}$$

*That is, the system is quadratically incrementally stable. The number* $\beta>0$ *depends only on the matrix* $\boldsymbol{X}$. *Furthermore, the system is globally exponentially convergent.*

Lyapunov-characterisations of incremental stability were first presented in [5] and ISS results for locally Lipschitz systems were published in [199].

PWA systems are here primarily viewed as local approximations of the nonlinear system (3.2) on a subset $\mathcal{V}$ of the state-space (see also [165]). For nonlinear systems (3.2) of the form

$$\begin{cases}\dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t))+\boldsymbol{B}\boldsymbol{u}(t)+\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)\\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t)\\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t),\end{cases} \tag{9.4}$$

Algorithm 9.1 constructs a Delaunay-partition $\mathcal{D}$ of $\mathcal{V}$ and a PWA system over $\mathcal{D}$ such that the model approximation error of the right-hand sides of (3.2) and (9.1) is smaller than a specified bound $\epsilon$

$$\forall\boldsymbol{x}\in\mathcal{V},\forall\boldsymbol{u}\in\mathbb{R}^m,\boldsymbol{d}\in\mathbb{R}^k:\|\boldsymbol{f}_a(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d})-(\boldsymbol{f}(\boldsymbol{x})+\boldsymbol{B}\boldsymbol{u}+\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d})\|\le\epsilon. \tag{9.5}$$

Various methods for the computation of the initial Delaunay partition and the refinement of a given partition are described elsewhere in detail [90].

**Example 9.1 (Piecewise affine model of a ship).** *The piecewise affine ship model is obtained by ignoring the input range limitations. Recall that, ignoring the input limitations, the state equation for the ship model (1.1)–(1.4) is of the form*

$$\dot{\boldsymbol{x}}(t)=\boldsymbol{f}(\boldsymbol{x}(t))+\boldsymbol{B}\boldsymbol{u}(t)+\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t).$$

*Therefore, a model approximation is only necessary for the autonomous part, which only depends on the state, and for which a piecewise affine approximation of arbitrary accuracy may be obtained at the cost of a very fine state space partition. The application of Algorithm 9.1 with* $\epsilon=1.4$ *provides a piecewise affine model of the form (9.1) on a partition with 196 simplices covering the hypercube* $-3\le v\le 3$, $-1\le w\le 1$, $-\pi\le r\le\pi$, *which is shown in Fig 9.2. The 196 parameter pairs* $(\boldsymbol{A}_i,\boldsymbol{a}_i)$ *are not explicitly written here. The model elements* $\boldsymbol{B}$, $\boldsymbol{B}_{\boldsymbol{d}}$, $\boldsymbol{C}$, *and* $\boldsymbol{C}_z$ *are the same as in the linear model (4.8), (4.9), because they involve no approximation.*

**Algorithm 9.1.** Error-bounded nonlinear system approximation by PWA system

**Require:** $f(x)$, $B$, $B_d$, $C$, $C_z$, $\epsilon$, $\mathcal{V}$

1: Generate initial Delaunay partition $\mathcal{D}_0 = \{\mathcal{S}_1, \mathcal{S}_2, \ldots\}$ of $\mathcal{V} \subset \mathbb{R}^n$, set $k = 0$
2: **repeat**
3: **for** $i = 0$ to $\dim(\mathcal{D}_k)$ **do**
4: Let $v_1, \ldots, v_{n+1}$ be the vertices of the simplex $\mathcal{S}_i$
5: Solve the system of equations $\begin{pmatrix} v_1^T & 1 \\ v_2^T & 1 \\ \vdots & \vdots \\ v_{n+1}^T & 1 \end{pmatrix} \begin{pmatrix} A_i^T \\ a_i^T \end{pmatrix} = \begin{pmatrix} f(v_1)^T \\ f(v_2)^T \\ \cdots \\ f(v_{n+1})^T \end{pmatrix}$ for $A_i$, $a_i$.
6: **end for**
7: Compute $e = \max_{x \in \mathcal{V}} \|A_j x + a_j - f(x)\|$
8: **if** $e > \epsilon$ **then**
9: Refine$(\mathcal{D}_k) \to \mathcal{D}_{k+1}$; $k = k + 1$
10: **end if**
11: **until** $e \le \epsilon$

**Result:** PWA system (9.1) that approximates the nonlinear system (9.4) within the error bound (9.5).

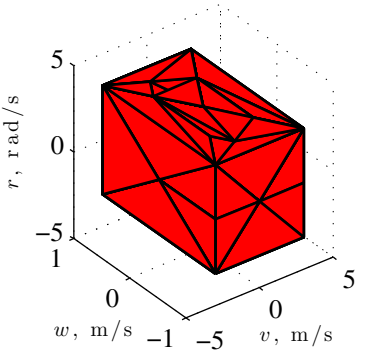


**Fig. 9.2** State-space partition into simplices for ship model.

*Fig. 9.3 shows the solutions of the nonlinear and two PWA models corresponding to the initial surge velocity $v(0) = 2$, initial sway velocity $w(0) = 0.1$, and initial yaw velocity $r(0) = -0.79$, the same case that was used to validate the linear model. The PWA models are based on 33 and 196 simplices. The figure shows clear improvements of PWA modelling compared to linear modelling. The surge, sway, and yaw velocities obtained using the PWA model all agree much better with those obtained using the nonlinear model than those obtained using the linear model (see Fig. 4.2 on page 58). The model accuracy improves with increasing number of simplices.*

*Fig. 9.4 compares the resulting ship positions and headings in the earth-fixed reference frame $(x, y)$ starting from the initial position $(x(0), y(0))^T = (0, 0)^T$ and the*

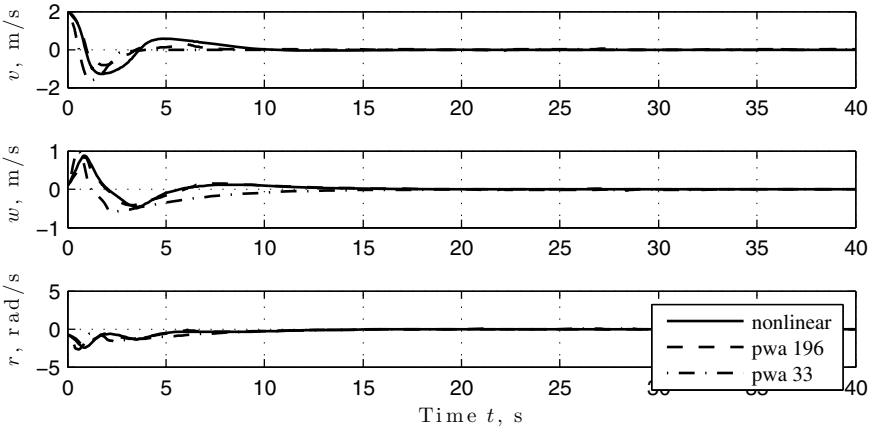


**Fig. 9.3** Comparison of ship responses obtained with a nonlinear and two PWA models (33 vs. 196 simplices) in terms of the surge, sway, and yaw velocities.

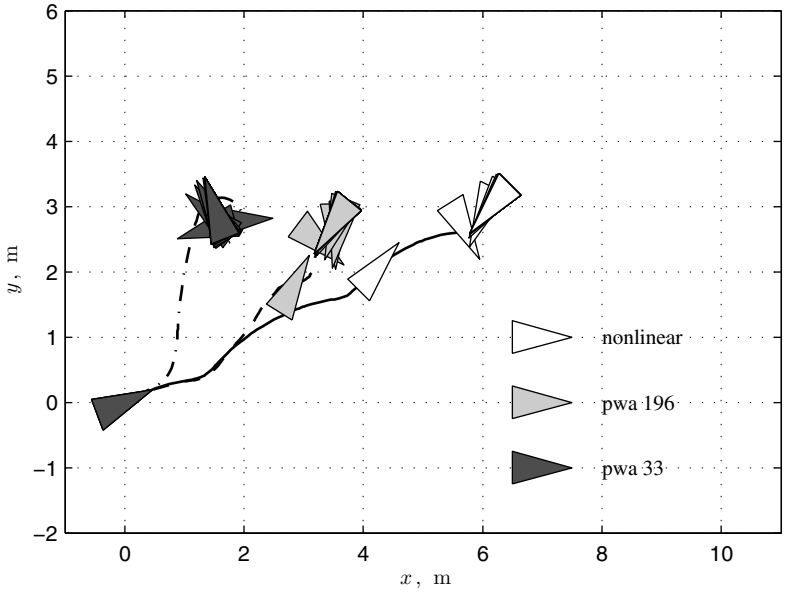


**Fig. 9.4** Comparison of ship responses obtained with a nonlinear and two PWA (33 vs. 196 simplices) models in absolute coordinates.

*initial heading $\psi(0) = 22.5°$. The heading is represented by an oriented triangular ship symbol shown every 5 seconds. Like the velocities, the position and heading obtained from the piecewise affine model correspond much better with those obtained with the nonlinear model than those obtained with the linearised model (see Fig. 4.3 on page 58). Notably, the PWA model based on 196 simplices correctly predicts the*

*rotation. Even the final heading agrees remarkably well with that obtained from the nonlinear model. The most obvious difference between the nonlinear and the piecewise affine models is the surge velocity, which causes a difference in the predicted translational motion. Summarising the model comparison, sufficiently complex PWA model are excellent approximations of nonlinear models. The accuracy comes at the price of high model complexity.*

## 9.2 Nominal Closed-Loop System and Assumptions

Throughout this part, the nominal controller $\Sigma_C$ for the nominal PWA plant (9.1) is a general nonlinear controller

$$\Sigma_C : \begin{cases} \dot{\boldsymbol{x}}_c(t) &= \boldsymbol{f}_c(\boldsymbol{x}_c(t), \boldsymbol{y}(t), \boldsymbol{r}(t)) \\ \boldsymbol{u}_c(t) &= \boldsymbol{h}_c(\boldsymbol{x}_c(t), \boldsymbol{y}(t), \boldsymbol{r}(t)), \end{cases} \qquad \boldsymbol{x}_c(0) = \boldsymbol{x}_{c0} \tag{9.6}$$

with the internal state $\boldsymbol{x}_c(t) \in \mathbb{R}^{n_c}$, the measurement $\boldsymbol{y}(t) \in \mathbb{R}^q$, the reference input $\boldsymbol{r}(t) \in \mathbb{R}^p$, and the control input $\boldsymbol{u}_c(t) \in \mathbb{R}^m$.

The nominal PWA plant (9.1) together with the controller (9.6) give rise to the nominal closed-loop system $(\Sigma_P, \Sigma_C)$.

The following assumptions are used for Part III of this monograph. The first assumption for the nominal closed-loop system is in place for solving the stability recovery problem.

**Assumption 9.2 (Stabilising nominal control).** *The nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$ that consists of the nominal PWA system (9.1) and the nominal controller (9.6) is ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$, and IOS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$ and the output $(\boldsymbol{x}, \boldsymbol{u}_c)$.*

The previous assumption means that, for bounded reference and disturbance inputs, the nominal closed-loop system responds with a bounded plant state and a bounded control input. The following assumption replaces Assumption 9.2 for solving the stability and tracking recovery problem.

**Assumption 9.3 (Stabilising and setpoint tracking nominal control).** *The nominal closed-loop system $\Sigma_L = (\Sigma_P, \Sigma_C)$ that consists of the nominal PWA system (9.1) with bounded measurement noise $\boldsymbol{n}_{\boldsymbol{y}}$ ($\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{n}_{\boldsymbol{y}}(t)$ where $\lim_{t\to\infty} \boldsymbol{n}_{\boldsymbol{y}}(t) = \boldsymbol{0}$) and the nominal controller (9.6) is ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{n}_{\boldsymbol{y}})$ and IOS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{n}_{\boldsymbol{y}})$ and the output $(\boldsymbol{x}, \boldsymbol{u}_c)$. Furthermore, constant reference commands $\boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$, $\bar{\boldsymbol{r}} \in \mathbb{R}^p$, are asymptotically tracked to precision $K \geq 0$ in the presence of arbitrary constant disturbances $\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t)$, $\bar{\boldsymbol{d}} \in \mathbb{R}^k$ and measurement noise $\boldsymbol{n}_{\boldsymbol{y}}$ with a constant steady-state control input $\bar{\boldsymbol{u}}_c \in \mathbb{R}^m$ in the sense that for all $\boldsymbol{x}_0$ and $\boldsymbol{x}_{c0}$*

$$\left\{\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t),\ \boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)\right\} \Rightarrow \begin{cases} \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{z}(t)\| \leq K \\ \lim_{t\to\infty} \boldsymbol{u}_c(t) = \bar{\boldsymbol{u}}_c. \end{cases}$$

The previous assumption is realistic in many cases and approaches for the tracking control of PWA systems have been recently reported in the literature [159, 228]. These approaches can be used to ensure or verify Assumption 9.3. The rejection of transient measurement noise is not restrictive but needed in the subsequent proofs.

## 9.3 Faults in Piecewise Affine Systems

In piecewise affine systems (9.1), faults are assumed not to affect the state-space partition, but only the system parameters as defined below.

**Definition 9.2 (Actuator faults in piecewise affine systems).** An *actuator fault* in a piecewise affine system is an event occurring at time $t_f$ that changes the nominal input matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ to the faulty input matrix $\boldsymbol{B}_f \in \mathbb{R}^{n\times m}$ of the same dimensions, and that changes the nominal affine terms $\boldsymbol{a}_i \in \mathbb{R}^n$ to the faulty affine terms $\boldsymbol{a}_{f,i} \in \mathbb{R}^n$ of the same dimensions. ◇

**Definition 9.3 (Sensor faults in piecewise affine systems).** A *sensor fault* in a piecewise affine system is an event occurring at time $t_f$ that changes the nominal measurement matrix $\boldsymbol{C} \in \mathbb{R}^{q\times n}$ to the faulty measurement matrix $\boldsymbol{C}_f \in \mathbb{R}^{q\times n}$ of the same dimensions. ◇

The fault event abruptly changes the nominal PWA system (9.1) to the faulty PWA system

$$\Sigma_{Pf}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}_i\boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \text{ for } \boldsymbol{x}_f(t) \in \Lambda_i,\ i \in \{1,\dots,r\} \\ \boldsymbol{y}_f(t) &= \boldsymbol{C}_f\boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) &= \boldsymbol{C}_z\boldsymbol{x}_f(t) \end{cases} \tag{9.7}$$
$$\boldsymbol{x}_f(0) = \boldsymbol{x}_0,$$

where the matrices $\boldsymbol{B}_f$, $\boldsymbol{C}_f$ and the family of vectors $\boldsymbol{a}_{f,i}$, $i \in \{1,\dots,r\}$, reflect the fault, whereas all other matrices remain unchanged. The parameter changes may be arbitrary as long as the obey the following assumption.

**Assumption 9.4.** *The right-hand side of the faulty system (9.7) is assumed to be a continuous function of $\boldsymbol{x}_f$, $\boldsymbol{u}_f$ and $\boldsymbol{d}$.*

The typical changes in the matrices $\boldsymbol{B}_f$ and $\boldsymbol{C}_f$ are the same as already discussed for linear systems in Section 4.3. The blockage of actuators $j \in \mathcal{A}$, $\mathcal{A} \subset \{1,\dots,m\}$, is typically modeled by means of a changed affine term

$$\boldsymbol{a}_{f,i} = \boldsymbol{a}_i + \sum_{j\in\mathcal{A}} \boldsymbol{b}_j\overline{u}_j,\ i \in \{1,\dots,r\} \tag{9.8}$$

and by means of corresponding zero columns in the matrix $\boldsymbol{B}_f$. In view of the typical model (9.8) of blocked actuators and the fact that the input matrices $\boldsymbol{B}$ and $\boldsymbol{B}_f$ are not mode-dependent, Assumption 9.4 is not restrictive.

A final assumption about the faulty system is made.

**Assumption 9.5 (Actuator blockage compensatability).** *The nominal PWA plant (9.1) and the faulty PWA plant (9.7) satisfy the condition*

$$\forall j \in \{1,\ldots,r\}: \; \boldsymbol{a}_j - \boldsymbol{a}_{f,j} \in \operatorname{im} \boldsymbol{B}_f. \tag{9.9}$$

This assumption means that the forcing input caused by blocked actuators can be compensated by the remaining, in other words functioning, actuators. The relaxation of this assumption is possible and is discussed after the presentation of the reconfiguration solution.

Table 9.1 summarises the expressiveness of the piecewise affine fault model. Clearly, the piecewise affine faulty system models are more expressive than the linear faulty system models. By comparison with Hammerstein-Wiener systems, however, reduced actuation ranges are not representable in PWA systems, since input limitations are not modelled in PWA systems at all. On the other hand, actuator blockage in arbitrary positions is readily representable within the piecewise affine model framework.

**Table 9.1** Technological faults representable by piecewise affine fault models.

| Technological fault | Representable | By model parameter |
|---|---|---|
| Changed actuator gain | ✓ | $\boldsymbol{B}_f$ |
| Changed nonlinear actuator characteristic | × | |
| Changed or reduced actuation range | × | |
| Actuator failure at the operating point | ✓ | $\boldsymbol{B}_f$ |
| Actuator failure off the operating point | ⓥ | $\boldsymbol{a}_{f,i}, \boldsymbol{B}_f$ |
| Changed sensor gain | ✓ | $\boldsymbol{C}_f$ |
| Changed nonlinear sensor characteristic | × | |
| Sensor failure | ✓ | $\boldsymbol{C}_f$ |

Legend: ✓: fully representable; ⓥ: exclusively representable in this system class; (✓): representable leaving the system class; ×: not representable.

**Example 9.2 (Piecewise affine model of a ship subject to faults).** *The PWA model of the faulty ship has the same model elements $\boldsymbol{B}_f$ and $\boldsymbol{C}_f$ as the linear model (4.19) of the faulty ship to represent the failed gyro sensor ($f_1$) and floating rudder ($f_3$). The blockage of the rudder ($f_2$) at the position $p$ is embedded into the affine model term $\boldsymbol{a}_{f,i}$ as follows:*

$$\boldsymbol{a}_{f,i} = \boldsymbol{a}_i + \boldsymbol{b}_3 p, \; i \in \{1,\ldots,r\}. \tag{9.10}$$

*The entire faulty ship is described by the PWA model*

$$\begin{cases} \dot{\boldsymbol{x}}_f(t) & = \boldsymbol{A}_i \boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f \boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}} \boldsymbol{d}(t) \text{ for } \boldsymbol{x}_f(t) \in \Lambda_i, \; i \in \{1,\ldots,r\} \\ \boldsymbol{y}_f(t) & = \boldsymbol{C}_f \boldsymbol{x}_f(t) \\ \boldsymbol{z}_f(t) & = \boldsymbol{C}_z \boldsymbol{x}_f(t). \end{cases} \tag{9.11}$$

*The fault $f_3$ meaning a floating rudder is obtained by the special case $p = 0$. The reduced actuation range of the left thruster ($f_4$) is not representable within the PWA modelling framework.*

**Example 9.3 (Piecewise affine model of the two-tank system).** *In this part, the running example consisting of the two-tank system defined in Chapter 1.5 is used. The two-tank system (1.14) has an input affine form similar to (9.4), with the difference that in the tanks system (1.14), the input matrix is a nonlinear function of the state. Its model is approximated by piecewise affine dynamics of the form (9.1), where the constant input matrix $\boldsymbol{B}$ is the Jacobian of the nonlinear vector field evaluated at the operating point $\boldsymbol{u}_0 = (0.5\ 0.5\ 0.8)^T$. The approximation gives rise to a PWA system with 22 regions. The vector fields of the nonlinear system and its PWA approximation along with the state-space partition are shown in Fig. 9.5 and 9.6.*

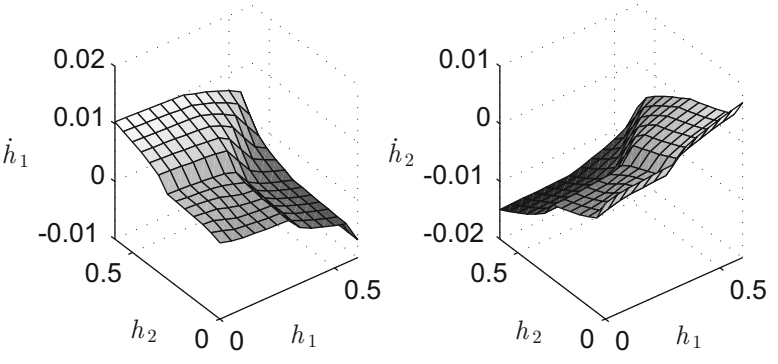


**Fig. 9.5** Nonlinear vector field of the two-tank system.

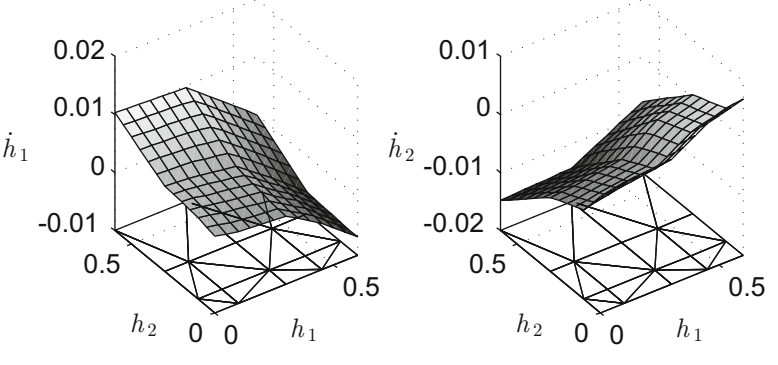


**Fig. 9.6** PWA vector field of the two-tank system.

*The response of the tanks model subject to these faults and reconfiguration using a linear virtual sensor and a linear virtual actuator is shown in Fig. 9.7. The sensor*

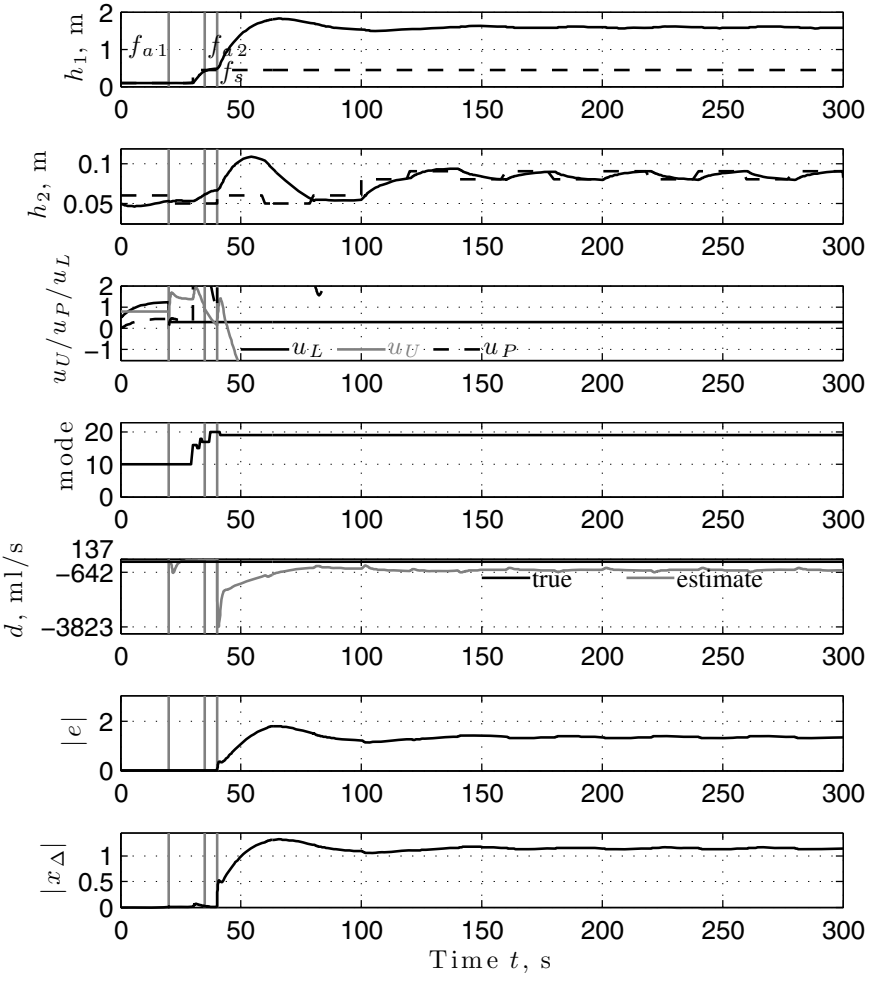


**Fig. 9.7** Response of reconfigured closed-loop tanks system based on linear virtual sensor and linear virtual actuator with periodic reference input.

*for the level $h_1$ fails at $t_{fs} = 40$ s, whereas the lower valve fails at $t_{fa1} = 20$ s and the upper valve degrades at $t_{fa2} = 35$ s. Although the system is stable due to the reconfiguration, it deviates so far from the reference trajectory that the left tank, whose height is 0.6 m, would overflow in reality. The disturbance estimate provided by the virtual sensor is very inaccurate. After large transients to −3823 ml/s, it settles at an steady-state value near −642 ml/s, which is considerably off the true value at −20 ml/s. The large estimation error and the large deviation from the nominal case are visible in the estimation error $\|\boldsymbol{e}\|$ and the difference state $\|\boldsymbol{x}_\Delta\|$ on the two lower axes.*

*While linear methods are adequate for the system of tanks if staying near the equilibrium of linearisation (see, for example, [171, 175]), this example shows that the linear methods are not sufficient when the system deviates considerably from an equilibrium, as it is the case in this startup procedure. This example emphasises the need for reconfiguration methods that provide global stability and tracking on the entire state-space. Such methods are presented in the next two chapters.*

## 9.4 Specific Reconfiguration Problems

The reconfiguration block (3.21) is a PWA system consisting of a PWA virtual actuator and a PWA virtual sensor. It can be written in the general form

$$\Sigma_R : \begin{cases} \dot{\boldsymbol{\xi}}(t) & = \boldsymbol{A}_{ri}\boldsymbol{\xi}(t) + \boldsymbol{a}_{ri} + \boldsymbol{B}_r\boldsymbol{u}_c(t) + \boldsymbol{E}_r\boldsymbol{y}_f(t) \text{ for } \boldsymbol{\xi}(t) \in \Lambda_{ri},\ i \in \{1,\dots,r_r\} \\ \boldsymbol{u}_f(t) & = \boldsymbol{C}_r\boldsymbol{\xi}(t) + \boldsymbol{D}_r\boldsymbol{u}_c(t) \\ \boldsymbol{y}_c(t) & = \boldsymbol{G}_r\boldsymbol{y}_f(t) + \boldsymbol{F}_r\boldsymbol{\xi}(t) \end{cases} \tag{9.12}$$
$$\boldsymbol{\xi}(0) = \boldsymbol{\xi}_0$$

and its state $\boldsymbol{\xi}(t) \in \mathbb{R}^{2n}$ has twice the number of states as the nominal and the faulty plant. The reconfiguration block is connected to the faulty plant (9.7) through their common signals $\boldsymbol{u}_f$ and $\boldsymbol{y}_f$ as well as to the nominal controller (9.6) through the common variable $\boldsymbol{u}_c$ and the interconnection $\boldsymbol{y}(t) = \boldsymbol{y}_c(t)$. The reconfiguration block state $\boldsymbol{\xi}$ resides in a state space that is partitioned into a collection of polyhedra $\Lambda_{ri}$, $i \in \{1,\dots,r_r\}$, whose union covers its entire state space: $\bigcup_{i=1}^{r_r} \Lambda_{ri} = \mathbb{R}^{2n}$. The mode of the reconfiguration block is determined exclusively by means of its state $\boldsymbol{\xi}$ and the polyhedra $\Lambda_{ri}$, $i \in \{1,\dots,r_r\}$. From a control point of view, it is of interest to at least recover the ISS property for the reconfigured closed-loop system (9.6), (9.7), (9.12). More precisely, the following problems are solved in this part.

**Problem 9.1 (Stability recovery for PWA systems).** Consider the nominal controller (9.6), the nominal PWA system (9.1), and the faulty PWA system (9.7). Find a reconfiguration block $\Sigma_R$ of the form (9.12) such that

$$\forall \Sigma_C :\ \{(\Sigma_P, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\}.$$

This problem concerns the stability recovery. The following problem extends the scope to setpoint tracking recovery.

**Problem 9.2 (Stable asymptotic setpoint tracking recovery for PWA systems).** Consider the nominal controller (9.6), the nominal PWA system (9.1), and the faulty PWA system (9.7). Find a reconfiguration block $\Sigma_R$ of the form (9.12) such that

$$\forall \Sigma_C :\ \{(\Sigma_P, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\} \Rightarrow \{(\Sigma_{Pf}, \Sigma_R, \Sigma_C) \text{ ISS w.r.t. } (\boldsymbol{r}, \boldsymbol{d})\},$$

and such that for constant disturbances $\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t)$, $\bar{\boldsymbol{d}} \in \mathbb{R}^k$, constant reference inputs $\boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$, $\bar{\boldsymbol{r}} \in \mathbb{R}^p$, and for all $\boldsymbol{x}_0 \in \mathbb{R}^n$, $\boldsymbol{x}_{c0} \in \mathbb{R}^{n_c}$, $\boldsymbol{\zeta}_0 \in \mathbb{R}^w$, it holds that

$$\left\{\limsup_{t\to\infty}\|\boldsymbol{r}(t)-\boldsymbol{z}(t)\|\le K\right\}\Rightarrow\left\{\limsup_{t\to\infty}\|\boldsymbol{r}(t)-\boldsymbol{z}_f(t)\|\le K\right\}.$$

The solutions to Problem 9.1 and Problem 9.2 are given in Chapter 10 and Chapter 11, respectively. The stability recovery for PWA systems was first described in [169]. The tracking recovery methods was first described in [170].

## 9.5 Bibliographic Notes on Piecewise Affine Systems

Piecewise affine systems are defined based on partitions of the state-space into polyhedra. In every polyhedron, the system is governed by a distinct affine (linear with offset) system equation. The motivation for studying PWA systems is at least twofold. Firstly, PWA systems are receiving wide attention due to the fact that the PWA framework initiated in [197] provides a way to describe dynamical systems exhibiting switching between a multitude of linear dynamical regimes, see also [35, 90]. The switching can be due to piecewise-linear characteristics such as dead-zone, saturation, hysteresis or relays. Secondly, PWA systems may result from piecewise linear approximations of complex nonlinear dynamics [90]. Finally, it has been shown in the discrete-time case that PWA systems are equivalent to other hybrid system models, such as mixed logical dynamical models, and linear complementarity systems under mild well-posedness assumptions [74]. General overviews of hybrid systems theory are available in [68, 118]. A good survey of switched linear systems is given in [105].

It has been recognised that many standard control-related analysis and synthesis problems for PWA systems are hard, in fact many of them are undecidable from a computer science point of view [24] in the general case. Therefore, special subclasses of PWA systems are frequently considered in the literature on stabilisation, state observation, and performance analysis of piecewise affine systems. On the other hand, general results for PWA systems tend to be conservative.

**Identification** methods for PWA models of dynamical systems has been studied based on a variety of ideas, an overview of which is available in [92]. In *bounded-error techniques* ideas from estimating a PWA autoregressive (PWARX) are combined with set-membership identification such that the satisfaction of a prescribed identification error bound is guaranteed [14]. A *clustering technique* for the measured data is used for PWARX model identification [56]. If the number of discrete modes are known a-priori, the identification task becomes easier [135]. Both the polyhedral state-space partition and the data of the PWARX model are estimated. In [137], the clustering step is facilitated by the introduction of a Gaussian mixture model and support vector classifiers for estimating boundary hyperplanes, which allows estimating the number of polyhedra required to represent the PWA model. An identification technique using PWA basis function models is described in [220]. In the *Bayesian approach*, the system parameters are treated as random variables whose probability density functions are iteratively updated based on the available measurements [95]. An *algebraic technique* that is provably correct in the noiseless case is described in [215]. A *mixed-integer optimisation* technique is

available in [182]. A relaxation of mixed-integer formulations into nonlinear nonconvex continuous-variable optimisation problems is described in [136].

**Stability analysis and controller synthesis** techniques based on piecewise quadratic Lyapunov functions are described for the continuous-time case in [165] (see also [231]) and for the discrete-time case in [54]. Verification and reachability analysis and verification of discrete-time PWA systems is studied in [15]. Optimal control of discrete-time PWA systems with polyhedral performance indices was studied in [11]. An $H_\infty$ optimal affine state-feedback control design has been proposed for uncertain PWA slab systems in [100] and for piecewise linear systems in [53]. In these methods, the state must be completely measurable and controller gains are scheduled based on the state. While these approaches reduce the conservatism associated with quadratic stability based on common Lyapunov functions, they are not applicable for the purposes of this monograph, because the state of the system is here assumed to be not completely measurable. An $H_\infty$ output-feedback control synthesis procedure is described in [20], which is based on dissipativity, storage functions, and common quadratric Lyapunov functions. The control problem for hybrid systems has been formulated as a mixed-integer quadratic program in [132, 133]. Model predictive control of continuous-time PWA systems was studied in [192], leading to a nonlinear nonconvex optimisation problem, and in [139] based on a receding-horizon technique with terminal cost and terminal set. Slightly outside the area of PWA systems, but closely related, are model predictive control techniques for nonlinear systems, e.g. [99].

**Observer design** for PWA system has been studied by several authors, and state observation for PWA systems is particularly difficult if the discrete mode is not known. The observability of discrete-time PWA systems was studied in [13], where it was shown that standard linear observability notions valid for each mode dynamics do not necessarily carry over to the PWA system. The case where the discrete mode is known is studied in [4]. Inference of the unknown discrete mode from discrete inputs and outputs was studied in [10]. The more challenging case of unknown discrete mode is studied for bimodal systems in [93] both for continuous and discontinuous dynamics. A moving-horizon state estimation scheme was proposed in [55], which is computationally demanding because a mixed-integer quadratic program has to be solved on line. For the general multimode case with unknown modes, common quadratic Lyapunov functions are the basis of presently available observer design methods [228]. Observer-based control has been studied in [75, 94, 228].

Similar to Hammerstein-Wiener systems, piecewise affine systems are practically very relevant, their control is nontrivial, and reconfigurable control of multimode PWA systems has been studied only by means of model predictive control techniques to the author's knowledge. Initial ideas for the fault diagnosis and control reconfiguration of bimodal piecewise affine systems with respect to stability recovery are reported in [138]. Necessary and sufficient reconfigurability conditions are likewise not to be expected and not achieved in this monograph. However, sufficient stabilisability conditions are provided in this monograph along with synthesis algorithms for recovering the nominal closed-loop stability and tracking properties.

# Chapter 10
# Stability Recovery after Actuator and Sensor Faults in Piecewise Affine Systems

**Abstract.** This chapter presents a reconfigurable control solution for piecewise affine systems subject to combined actuator and sensor faults based on the fault-hiding idea. The approach recovers the input-to-state stability property for the reconfigured closed-loop system, and it is shown to be robust against uncertainties in the piecewise affine model of the faulty plant.

## 10.1 Piecewise Affine Virtual Actuator and Virtual Sensor

For the combined occurrence of actuator faults and sensor faults ($\boldsymbol{a}_{f,i} \neq \boldsymbol{a}_i$, $\boldsymbol{B}_f \neq \boldsymbol{B}$, $\boldsymbol{C}_f \neq \boldsymbol{C}$), the reconfiguration block $\Sigma_R$ defined in Equation (9.12) is realised by the interconnection of a piecewise affine virtual sensor $\Sigma_S$ and a piecewise affine virtual actuator $\Sigma_A$, which are defined as follows (Fig. 10.1).

**Definition 10.1 (Piecewise affine virtual sensor).** The *piecewise affine virtual sensor* is the dynamical system

$$\Sigma_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= (\boldsymbol{A}_i - \boldsymbol{L}\boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{L}\boldsymbol{y}_f(t) \text{ for } \hat{\boldsymbol{x}}_f(t) \in \Lambda_i, \\ & \quad i \in \{1,\ldots,r\} \\ \hat{\boldsymbol{y}}_c(t) &= \boldsymbol{y}_f(t) + (\boldsymbol{C} - \boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) \end{cases} \tag{10.1}$$

with the initial condition $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$. ◇

The PWA virtual sensor contains a model of the faulty plant (9.7) augmented by output error injection. Its mode $i$ is completely determined based on the virtual sensor state $\hat{\boldsymbol{x}}_f$ and requires no information about the mode of the observed faulty PWA system.

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 157–168.
springerlink.com

**Definition 10.2 (Piecewise affine virtual actuator).** The *piecewise affine virtual actuator* is the dynamical system

$$\Sigma_A : \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) & = \boldsymbol{A}_j \tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j + \boldsymbol{B}\boldsymbol{u}_c(t) \text{ for } \tilde{\boldsymbol{x}}(t) \in \Lambda_j,\ j \in \{1,\ldots,r\} \\ \boldsymbol{y}_c(t) & = \hat{\boldsymbol{y}}_c(t) + \boldsymbol{C}\boldsymbol{x}_\Delta(t) \\ \boldsymbol{u}_f(t) & = \boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{B}_f^+ \boldsymbol{a}_{\Delta,j} \end{cases} \tag{10.2}$$

with the initial condition $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_{f0}$, where $\boldsymbol{x}_\Delta(t) = \tilde{\boldsymbol{x}}(t) - \hat{\boldsymbol{x}}_f(t)$ is in accordance with Equation (3.35), and where

$$\boldsymbol{a}_{\Delta,j} \triangleq \boldsymbol{a}_j - \boldsymbol{a}_{f,j} \tag{10.3}$$

is the difference between nominal and faulty affine terms. ◇

The reconfiguration block $\Sigma_R = (\Sigma_S, \Sigma_A)$ contains a PWA virtual sensor (10.1) with the state $\hat{\boldsymbol{x}}_f(t)$ at time $t$, and a PWA virtual actuator (10.2) with the state $\tilde{\boldsymbol{x}}(t)$ at time $t$. The PWA virtual sensor (10.1) contains an observer for the faulty plant $\Sigma_{Pf}$. The PWA virtual actuator (10.2) contains a reference model for the desired (fault-free) plant behavior $\Sigma_P$, and feedback to stabilise the faulty plant. The reconfiguration block (9.12) is initialised with $\boldsymbol{\zeta}(0) = (\hat{\boldsymbol{x}}_f(0)^T,\ \tilde{\boldsymbol{x}}(0)^T)^T = \boldsymbol{\zeta}_0 = (\hat{\boldsymbol{x}}_{f0}^T,\ \hat{\boldsymbol{x}}_{f0}^T)^T$ (Fig. 10.1). In other words, the PWA virtual sensor $\Sigma_S$ and the PWA virtual actuator $\Sigma_A$ are initialised at equal values at reconfiguration time $t = 0$.

Note that the reconfiguration block defined by the PWA virtual sensor and the PWA virtual actuator satisfies almost all inactivity conditions (3.22), (3.23) prior to reconfiguration time, where $\boldsymbol{C}_f = \boldsymbol{C}$ and $\boldsymbol{B}_f = \boldsymbol{B}$, and conseqently $\hat{\boldsymbol{y}}_c = \boldsymbol{y}_f$ and $\boldsymbol{y}_c = \boldsymbol{y}_f$. Furthermore, $\boldsymbol{a}_{\Delta,j} = \boldsymbol{0}$ holds prior to faults. The further inactivity condition $\boldsymbol{u}_f = \boldsymbol{u}_c$ must be enforced separately before the appearance of faults.

The affine input $\boldsymbol{B}_f^+ \boldsymbol{a}_{\Delta,j}$ compensates the bias introduced by the difference between the affine terms of the nominal and the faulty plant, which arises, for example, from blocking actuators as discussed before. Consider now the blockage of some actuators whose column indices in the matrix $\boldsymbol{B}$ are collected in the set $J$. In accordance with Equation (9.8), the difference (10.3) between the nominal and faulty affine term

$$\boldsymbol{a}_\Delta = \sum_{k \in J} \boldsymbol{b}_k \overline{u}_k \tag{10.4}$$

is mode-independent. The desired compensation is successful if and only if Condition (9.9) is satisfied, which is true by Assumption 9.5.

The observation error system $\Sigma_e$ and difference system $\Sigma_\Delta$ are governed by the following differential equations, which are easily obtained from the definitions of $\boldsymbol{e}$ (Equation (3.39)) and $\boldsymbol{x}_\Delta$ (Equation (3.35)) as well as from Assumption 9.5 by means of straightforward calculations:

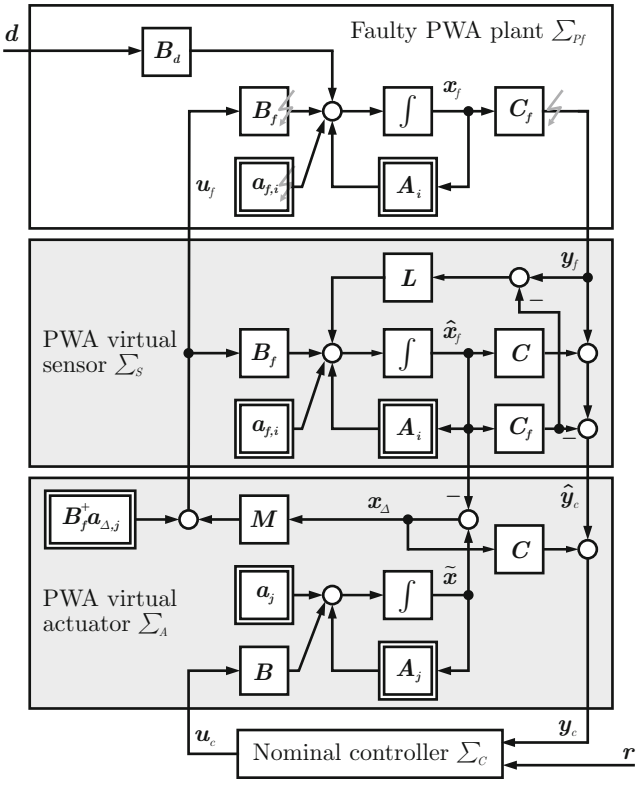


**Fig. 10.1** PWA virtual actuator and virtual sensor for stability recovery after the combined occurrence of actuator and sensor faults.

$$\Sigma_e : \begin{cases} \dot{\boldsymbol{e}}(t) & = \boldsymbol{k}_e(\boldsymbol{x}_f(t)+\boldsymbol{e}(t)) - \boldsymbol{k}_e(\boldsymbol{x}_f(t)) - \boldsymbol{B_d}\boldsymbol{d}(t) \end{cases} \tag{10.5}$$

$$\text{where } \boldsymbol{k}_e(\boldsymbol{\xi}) \triangleq (\boldsymbol{A}_i - \boldsymbol{L}\boldsymbol{C}_f)\boldsymbol{\xi} + \boldsymbol{a}_{f,i},\ \boldsymbol{\xi} \in \Lambda_i,\ i \in \{1,\ldots,r\}, \tag{10.6}$$

$$\Sigma_\Delta : \begin{cases} \dot{\boldsymbol{x}}_\Delta(t) & = \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t)) - \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t)) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) \end{cases} \tag{10.7}$$

$$\text{where } \boldsymbol{k}_\Delta(\boldsymbol{\eta}) \triangleq (\boldsymbol{A}_j - \boldsymbol{B}_f\boldsymbol{M})\boldsymbol{\eta} + \boldsymbol{a}_{f,j},\ \boldsymbol{\eta} \in \Lambda_j,\ j \in \{1,\ldots,r\}. \tag{10.8}$$

Assumption 9.5 is only needed to rearrange the difference system in the form (10.7), the observation error (10.5) does not depend on that assumption. The PWA virtual actuator may be interpreted as an approach to match the reconfigured plant behavior to the nominal plant behavior. The feedback gains $\boldsymbol{L} \in \mathbb{R}^{n\times q}$ and $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ will be designed to stabilise the observation error $\boldsymbol{e}$ as well as the difference system state $\boldsymbol{x}_\Delta$.

It is first shown that the reconfiguration block (10.1), (10.2) in combination with the faulty plant (9.7) satisfies the weak fault-hiding goal.

**Lemma 10.1 (Weak fault-hiding).** *The reconfigured plant $\Sigma_{Pr} = (\Sigma_{Pf}, \Sigma_S, \Sigma_A)$ formed by the faulty PWA plant (9.7), the PWA virtual sensor (10.1), and the PWA virtual actuator (10.2) satisfies the weak fault-hiding goal.*

*Proof.* Based on Assumption 9.5, the model of the reconfigured plant is given by the equations (index $j$ defined by $\tilde{\boldsymbol{x}}(t) \in \Lambda_j$)

$$\begin{pmatrix} \dot{\tilde{\boldsymbol{x}}}(t) \\ \dot{\boldsymbol{e}}(t) \\ \dot{\boldsymbol{x}}_{\Delta}(t) \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_j \tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j \\ \boldsymbol{k}_e(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)) - \boldsymbol{k}_e(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t)) \\ \boldsymbol{k}_{\Delta}(\tilde{\boldsymbol{x}}(t)) - \boldsymbol{k}_{\Delta}(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)) + \boldsymbol{L}\boldsymbol{C}_f \boldsymbol{e}(t) \end{pmatrix} + \begin{pmatrix} \boldsymbol{B} \\ \boldsymbol{0} \\ \boldsymbol{B} \end{pmatrix} \boldsymbol{u}_c(t) - \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{B}_{\boldsymbol{d}} \\ \boldsymbol{0} \end{pmatrix} \boldsymbol{d}(t), \quad (10.9)$$

$$\boldsymbol{y}_c(t) = \begin{pmatrix} \boldsymbol{C} & -\boldsymbol{C}_f & \boldsymbol{0} \end{pmatrix} \begin{pmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{e}(t) \\ \boldsymbol{x}_{\Delta}(t) \end{pmatrix}, \begin{pmatrix} \tilde{\boldsymbol{x}}(0) \\ \boldsymbol{e}(0) \\ \boldsymbol{x}_{\Delta}(0) \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{x}}_{f0} \\ \hat{\boldsymbol{x}}_{f0} - \boldsymbol{x}_0 \\ \boldsymbol{0} \end{pmatrix}. \quad (10.10)$$

This model shows that the dynamical equation for the reference state $\tilde{\boldsymbol{x}}$ is decoupled from the observation error $\boldsymbol{e}$ and the difference state $\boldsymbol{x}_{\Delta}$. The output $\boldsymbol{y}_c$ depends on $\tilde{\boldsymbol{x}}$ and $\boldsymbol{e}$, where the observation error $\boldsymbol{e}$ is autonomous with respect to the control input $\boldsymbol{u}_c$ and only driven by the disturbance $\boldsymbol{d}$. Weak fault hiding (Definition 3.6) is achieved by the matching initialisation $\hat{\boldsymbol{x}}_{f0} = \boldsymbol{x}_0$, and the nominal I/O behaviour is recovered for $\boldsymbol{d} \equiv \boldsymbol{0}$. ■

The matching initialisation $\hat{\boldsymbol{x}}_{f0} = \boldsymbol{x}_0$ is in general practically not achievable, because the initial condition $\boldsymbol{x}_0$ is often not completely measurable. Furthermore, the disturbance does not appear in the output $\boldsymbol{y}_c$. However, the stability recovering solution to Problem 9.1 described in the next section is also achieved for inaccurate initialisation and mismatch in the disturbance behaviour. Two ways of approximately matching initialisation are opened by running the PWA virtual sensor based on the nominal PWA model (9.1) before the appearance of the fault.

1. If the combined diagnosis and reconfiguration delay is small with respect to the plant time constants, then the error made by initialising with the last state estimate will be small.
2. If the diagnosis delay is large, then it is useful to first design and update the PWA virtual sensor, wait for convergence of the observation error for a sufficiently long period of time, and to finally update and initialize the PWA virtual actuator with the virtual sensor state valid at that time.

The following section presents the main stability result and associated synthesis methods for the PWA virtual sensor and the PWA virtual actuator.

## 10.2 Main Stability Result

In this section, the solution to Problem 9.1 is given. Namely, sufficient conditions for the global ISS properties of the observation error and difference system are provided in two lemmas, and it is shown that these conditions also imply the global ISS of the reconfigured closed-loop system.

**Lemma 10.2 (Input-to-state stability of the observation error).** *Consider the faulty PWA system (9.7), and suppose that Assumption 9.4 holds. If there exist matrices $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$, $\boldsymbol{X}_s = \boldsymbol{X}_s^T \succ 0$, and $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$ that satisfy the LMIs*

$$\boldsymbol{X}_s\boldsymbol{A}_j + \boldsymbol{A}_j^T\boldsymbol{X}_s - \boldsymbol{Y}_s\boldsymbol{C}_f - \boldsymbol{C}_f^T\boldsymbol{Y}_s^T \prec 0,\ j = 1,\dots,r, \tag{10.11}$$

*then the system (10.1) with the gain $\boldsymbol{L} \triangleq \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$ is an observer for the faulty system (9.7) with 0-GES error dynamics for $\boldsymbol{d} \equiv \boldsymbol{0}$. The observation error (3.39) satisfies the dynamics (10.5), (10.6), and all solutions $\boldsymbol{e}(t)$ of the undisturbed system (10.5), (10.6) satisfy the relation*

$$\|\boldsymbol{e}(t)\| \le ce^{-at}\|\boldsymbol{e}(0)\|,\ t \in [0,\infty), \tag{10.12}$$

*where the real numbers $c > 0$ and $a > 0$ depend only on $\boldsymbol{X}_s$ and $\boldsymbol{Y}_s$. Furthermore, the observation error (10.5) is ISS w.r.t. the disturbance input $\boldsymbol{d}$.*

*Proof.* The case without disturbance is equivalent to the case considered in [158, 228]. Note that the function (10.6) is continuous in its arguments by construction. To show ISS of the error dynamics (10.5), (10.6) w.r.t. the disturbance input $\boldsymbol{d}$, an ISS-Lyapunov function $V(\boldsymbol{e}) = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{X}\boldsymbol{e}$ is constructed. By using Theorem 9.1 while observing that the error dynamics (10.5), (10.6) is continuous, one directly obtains for some $a > 0$, $b > 0$, $\theta \in (0,1)$ that

$$\begin{aligned}
\dot{V}(\boldsymbol{e}) = \boldsymbol{e}^T\boldsymbol{X}\dot{\boldsymbol{e}} &= \boldsymbol{e}^T\boldsymbol{X}(\boldsymbol{k}_e(\boldsymbol{x}_f + \boldsymbol{e}) - \boldsymbol{k}_e(\boldsymbol{x}_f) - \boldsymbol{B}_d\boldsymbol{d}) \\
&\le -a\boldsymbol{e}^T\boldsymbol{X}\boldsymbol{e} - \boldsymbol{e}^T\boldsymbol{X}\boldsymbol{B}_d\boldsymbol{d} \\
&\le -b\|\boldsymbol{e}\|^2 + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\boldsymbol{B}_d\|\cdot\|\boldsymbol{d}\| \\
&= -(1-\theta)b\|\boldsymbol{e}\|^2 - \theta b\|\boldsymbol{e}\|^2 + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\boldsymbol{B}_d\|\cdot\|\boldsymbol{d}\| \\
&\le -(1-\theta)b\|\boldsymbol{e}\|^2 \text{ if } \|\boldsymbol{e}\| \ge \frac{\|\boldsymbol{X}\|\cdot\|\boldsymbol{B}_d\|}{\theta b}\|\boldsymbol{d}\|,
\end{aligned}$$

which is a Lyapunov characterisation of the ISS property [199]. ∎

**Lemma 10.3 (Input-to-state stability of the difference system).** *Consider the faulty PWA system (9.7) and suppose that Assumptions 9.5 and 9.4 hold. If there exist matrices $\boldsymbol{X}_a \in \mathbb{R}^{n\times n}$, $\boldsymbol{X}_a = \boldsymbol{X}_a^T \succ 0$, and $\boldsymbol{Y}_a \in \mathbb{R}^{m\times n}$ that satisfy the LMIs*

$$\boldsymbol{A}_i\boldsymbol{X}_a + \boldsymbol{X}_a\boldsymbol{A}_i^T - \boldsymbol{B}_f\boldsymbol{Y}_a - \boldsymbol{Y}_a^T\boldsymbol{B}_f^T \prec 0,\ i = 1,\dots,r, \tag{10.13}$$

*then the difference system (10.7), (10.8) of the virtual actuator (10.2) with the gain $\boldsymbol{M} \triangleq \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$ is 0-GES for $\boldsymbol{u}_c, \boldsymbol{e} \equiv \boldsymbol{0}$. In other words, every solution $\boldsymbol{x}_\Delta(t)$ of the unforced difference system (10.7), (10.8) (i.e. $\boldsymbol{u}_c, \boldsymbol{e} \equiv \boldsymbol{0}$) satisfies the relation*

$$\|\boldsymbol{x}_\Delta(t)\| \le ce^{-at}\|\boldsymbol{x}_\Delta(0)\|,\ t \in [0,\infty), \tag{10.14}$$

*where the real numbers $c > 0$ and $a > 0$ depend only on $\boldsymbol{X}_a$ and $\boldsymbol{Y}_a$. Furthermore, the difference system (10.7) is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{e})$.*

*Proof.* Observing that the difference system (10.7), (10.8) is continuous by construction, it is globally exponentially stable for $\boldsymbol{u}_c(t) = \boldsymbol{e}(t) = \boldsymbol{0}$ by Theorem 9.1 and ISS w.r.t. its input $(\boldsymbol{u}_c, \boldsymbol{e})$ if the condition

$$\tilde{\boldsymbol{X}}(\boldsymbol{A}_j - \boldsymbol{B}_f\boldsymbol{M}) + (\boldsymbol{A}_j - \boldsymbol{B}_f\boldsymbol{M})^T\tilde{\boldsymbol{X}} \prec 0,\ \tilde{\boldsymbol{X}} = \tilde{\boldsymbol{X}}^T \succ 0$$

is satisfied for all $j = 1,\ldots,r$, which is equivalent to Condition (10.13) after pre- and postmultiplication with $\tilde{\boldsymbol{X}}^{-1}$, reordering, and linearising changes of variables $\boldsymbol{X}_a = \tilde{\boldsymbol{X}}^{-1}$ and $\boldsymbol{Y}_a = \boldsymbol{M}\boldsymbol{X}_a$. The exponential decay rate of the initial state follows from Theorem 9.1. It is now explicitly shown that the difference system is indeed ISS w.r.t. $(\boldsymbol{u}_c, \boldsymbol{e})$. Consider the quadratic function $V(\boldsymbol{x}_\Delta) = \frac{1}{2}\boldsymbol{x}_\Delta^T\boldsymbol{P}\boldsymbol{x}_\Delta$. Using Theorem 9.1, its derivative along solutions of (10.7), (10.8) satisfies

$$\begin{aligned}\dot{V}(\boldsymbol{x}_\Delta) &= \boldsymbol{x}_\Delta^T\boldsymbol{P}\Big(\boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}) - \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}} - \boldsymbol{x}_\Delta) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e} + \boldsymbol{B}\boldsymbol{u}_c\Big)\\ &\leq -a\boldsymbol{x}_\Delta^T\boldsymbol{P}\boldsymbol{x}_\Delta + \boldsymbol{x}_\Delta^T\boldsymbol{P}\boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e} + \boldsymbol{x}_\Delta^T\boldsymbol{P}\boldsymbol{B}\boldsymbol{u}_c,\end{aligned}$$

where $a > 0$ is a constant. This inequality is readily transformed into a Lyapunov characterisation of ISS (see the proof of Lemma 10.2), and consequently the difference system (10.7), (10.8) is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{e})$. ■

Combining Lemma 10.1, Lemma 10.2 and Lemma 10.3, the main result that solves Problem 9.1 is presented. Its proof is technical and available in the appendix.

**Theorem 10.1 (Reconfigured closed-loop stability recovery).** *Suppose that the Assumptions 9.1, 9.2, 9.5, and 9.4 are satisfied, and suppose that the LMIs (10.11) and (10.13) are feasible. Then, the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_S, \Sigma_A, \Sigma_C)$ consisting of the faulty PWA system (9.7), the controller (9.6), the PWA virtual sensor (10.1), and the PWA virtual actuator (10.2) is globally ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$.*

*Proof.* See Appendix D, page 265.

*Remark 10.1 (Role of fault-hiding).* The obtained stability results do not depend on the accuracy of the guessed initial conditions. In other words, the results are valid whether or not the initial condition of the reconfiguration block satisfies $\hat{\boldsymbol{x}}_{f0} = \boldsymbol{x}_0$ and $\tilde{\boldsymbol{x}}_0 = \boldsymbol{x}_0$, as it is clear from the proof of Theorem 10.1 by considering that $\hat{\boldsymbol{x}}_{f0}$ only affects $\boldsymbol{e}(0)$. The relevance of the weak fault-hiding goal (Lemma 10.1) consists in re-introducing the nominal closed-loop dynamics into the reconfigured closed-loop system, see the proof of Theorem 10.1. ○

*Remark 10.2 (Non-compensable actuator blockage).* Assumption 9.5 can be easily relaxed as follows. In the case where the actuator blockage cannot be compensated, the difference system (10.7) is augmented by an added term $\Sigma_\Delta:\ \dot{\boldsymbol{x}}_\Delta(t) = \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t)) - \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t)) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) + (\boldsymbol{I} - \boldsymbol{B}_f\boldsymbol{B}_f^+)\boldsymbol{a}_\Delta g(t)$ where $g(t) = 1$, which acts

like a constant additive input to the difference system. The difference system is thus also ISS with respect to a new fictitious input $g$ that acts on the difference system through the structure $(\boldsymbol{I} - \boldsymbol{B}_f \boldsymbol{B}_f^+)\boldsymbol{a}_\Delta$, to which a constant input $g(t) = 1$ is applied. This implies that the system is input-to-state practically stable (ISpS) w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d})$ in the sense that convergence to a neighbourhood of the origin is achieved in the absence of further inputs, instead of convergence to the origin. The size of the achieved set depends on the gain from the input $g$ to the state $\boldsymbol{x}_\Delta$. ∘

The design procedure of the reconfiguration block for achieving stability recovery is summarised in Algorithm 10.1. The steps 1-4 describe the nominal closed-loop operation before any faults occur. Note that the guessed initial condition $\hat{\boldsymbol{x}}_f(t_0)$ is not required to be accurate. Once faults are detected in step 5, the virtual sensor and virtual actuator design activates in steps 6-11, where the gains $\boldsymbol{L}$ and $\boldsymbol{M}$ are determined. After the gain calculations are finished, the reconfigured closed-loop system is executed in step 12, starting at reconfiguration time $t = 0$.

**Algorithm 10.1.** Stability recovering PWA virtual sensor and virtual actuator synthesis

**Require:** PWA plant model $\boldsymbol{A}_i$, $\boldsymbol{a}_i$, $\boldsymbol{B}$, $\boldsymbol{C}$, $i \in \{1, \ldots, r\}$, initial time $t_0 < 0$, guessed initial condition $\hat{\boldsymbol{x}}_{f0}$

1: Initialise the nominal closed-loop system (9.1), (9.6), (10.1), (10.2), with $\boldsymbol{C}_f = \boldsymbol{C}$, $\boldsymbol{B}_f = \boldsymbol{B}$, $\boldsymbol{a}_{f,i} = \boldsymbol{a}_i$, $\boldsymbol{L} = \boldsymbol{0}$, $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, $\boldsymbol{x}_c(t_0) = \boldsymbol{x}_{c0}$, $\hat{\boldsymbol{x}}_f(t_0) = \hat{\boldsymbol{x}}_{f0}$, $\tilde{\boldsymbol{x}}(t_0) = \hat{\boldsymbol{x}}_{f0}$. Set the virtual actuator inactive by setting $\boldsymbol{u}_f(t) = \boldsymbol{u}_c(t)$.
2: Solve the LMI (10.11) with $\boldsymbol{C}_f = \boldsymbol{C}$ and compute a stabilising virtual sensor gain $\boldsymbol{L} \triangleq \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$, update PWA virtual sensor (10.1)
3: **repeat**
4: Run nominal closed-loop system
5: **until** actuator or sensor fault $f$ detected and isolated
6: Construct fault model $\boldsymbol{a}_{f,i}$, $\boldsymbol{B}_f$, $\boldsymbol{C}_f$ and update the PWA virtual sensor (10.1) and the PWA virtual actuator (10.2)
7: Solve the LMIs (10.11) and (10.13) for $\boldsymbol{X}_s$, $\boldsymbol{Y}_s$, $\boldsymbol{X}_a$, $\boldsymbol{Y}_a$
8: Compute $\boldsymbol{L} \triangleq \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$ and $\boldsymbol{M} \triangleq \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$
9: Update the PWA virtual sensor (10.1) with $\boldsymbol{L}$
10: Wait for PWA virtual sensor to converge for specified time interval
11: Update and activate the PWA virtual actuator (10.2) with $\boldsymbol{M}$ and initialise $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_f(0)$
12: Run reconfigured closed-loop system (9.7), (9.6), (10.1), (10.2)

**Result:** Globally ISS reconfigured closed-loop system.

**Example 10.1 (Stability recovering PWA virtual sensor and actuator synthesis for the two-tank system).** *A PWA virtual sensor and a PWA virtual actuator are designed for the two-tank system in order to reconfigure the closed-loop system after*

*the occurrence of the sensor fault $f_s$ and the valves failure $f_{a1}$ and degradation $f_{a2}$ in order to recover closed-loop stability.*

*The response of the tanks model subject to these faults is shown in Fig. 10.2. The sensor for the level $h_1$ fails at $t_{fs} = 40$ s, whereas the lower valve fails at $t_{fa1} = 20$ s and the upper valve degrades at $t_{fa2} = 35$ s. Due to the reconfiguration, it can be seen that the system states (solid in upper two axes) are stable and stay close to the reference trajectories (dashed in upper two axes). However, the disturbance appearing at $t = 65$ s prevents exact tracking (5th axis from above), since it causes an offset in the state estimate inside the virtual sensor. It can be seen that the PWA*

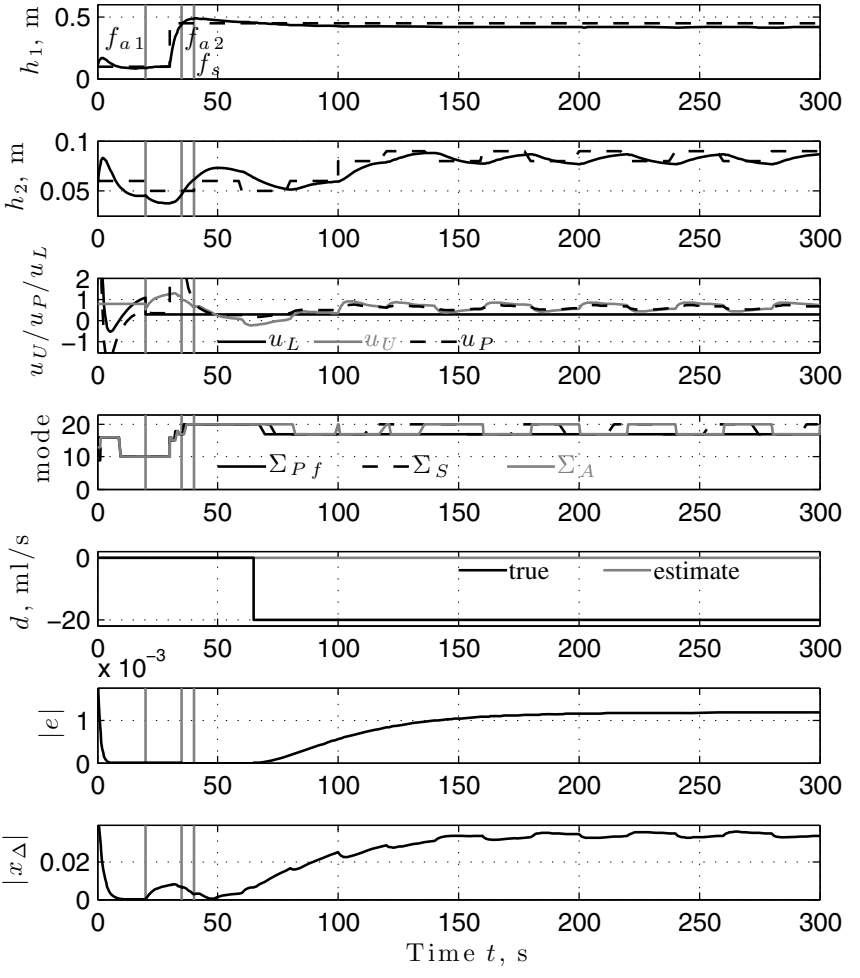


**Fig. 10.2** Response of reconfigured closed-loop tanks system based on stabilising PWA virtual sensor and stabilising PWA virtual actuator with periodic reference input.

*virtual sensor and the PWA plant are in different modes frequently and for extended periods of time. The PWA virtual actuator compensates the blocked lower valve by moving the upper valve to a compensating offset position. Although closed-loop stability is recovered, the result is not yet satisfactory since the tracking property is not recovered. In particular, there remains a steady-state offset between reference and true output for the level $h_1$, and the signal $h_2$ is lagging behind its reference signal $r_2$.*

## 10.3 Reconfigurability Considerations

In the linear case, the conditions (10.11) and (10.13) reduce to standard Lyapunov inequalities, which have feasible solutions if and only if the pairs $(\boldsymbol{C}_f, \boldsymbol{A})$ and $(\boldsymbol{A}, \boldsymbol{B}_f)$ are detectable and stabilisable, respectively. In the linear case, the existence of a quadratic Lyapunov function for the faulty system is hence necessary and sufficient for its stabilisability. In the case of PWA systems, the LMIs (10.11) and (10.13) represent common quadratic Lyapunov functions for the PWA observation error and the PWA difference system. This relationship is emphasised by the fact that the gains $\boldsymbol{L}$ and $\boldsymbol{M}$ are not mode-dependent. It is known that not every stabilisable PWA system admits a common quadratic Lyapunov function [90].

If the relevant LMIs are infeasible, then a stabilising PWA virtual sensor and PWA virtual actuator scheme might exist, but it cannot be found using the sufficient stability conditions presented in this chapter. This problem appears to be fundamentally unavoidable, since the problem of deciding whether all trajectories of a given PWA system are bounded is undecidable [24].

At first glance, it might seem that a reduction of this conservatism is achievable by seeking continuous piecewise quadratic Lyapunov functions instead of common quadratic Lyapunov functions [91, 105]. However, in those works, sufficient conditions for stability (and stabilisation) in terms of the existence of piecewise quadratic Lyapunov functions have been obtained for *equilibria* of PWA systems (all solutions converging to the origin). In the approach presented in this chapter, the stability of *time-varying solutions* of certain PWA systems is required (such as the observation error and difference systems), and such stability properties are studied using the concepts of convergence/incremental stability. No such characterisation of incremental stability or convergence for PWA systems in terms of piecewise quadratic Lyapunov functions exists to date to the knowledge of the author.

Suppose that, nevertheless, a piecewise quadratic Lyapunov function and associated mode-dependent gains $\boldsymbol{L}_i$ and $\boldsymbol{M}_i$ exist. The implementation in the case of the virtual actuator requires

1. knowledge of $\tilde{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}_f$ (which is not problematic), in order to
2. schedule the gain $\boldsymbol{M}_i$ depending on both $\tilde{\boldsymbol{x}}$ and $\hat{\boldsymbol{x}}_f$, therefore the number of possible gains is $r^2$, if $r$ denotes the number of polyhedra covering the state space.

In the case of the virtual sensor, which is essentially an observer, the same considerations hold. Aspect 1 is a severe problem in this case, since the plant state is unknown, and the difficulty in observer design for piecewise affine systems consists in the need to stabilise every possible mixed mode of the observation error. Sufficient stability conditions that are less conservative than those of Lemma 10.2 have so far only been found for certain special cases, such as bimodal systems [94].

## 10.4 Robustness against Piecewise Affine Model Uncertainties

In the previous sections, it has been assumed that the faulty plant, the PWA virtual sensor, and the PWA virtual actuator are PWA systems. The motivation for studying this class of systems arises, however, from their ability to well approximate input-affine nonlinear systems. In reality, the faulty system is more accurately represented by a nonlinear model, whereas the reconfiguration blocks are implemented based on a PWA system model. The question arises how robust the reconfiguration scheme is against approximation errors of the PWA model. These properties of the closed-loop reconfiguration scheme are studied in this section, answering the question about the robustness of this reconfiguration scheme.

For the analysis, assume that the faulty nonlinear system is an input-affine system of the form

$$\Sigma_{Pf,NL}: \left\{\dot{\boldsymbol{x}}_f(t) = \boldsymbol{f}(\boldsymbol{x}_f(t)) + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t), \right. \tag{10.15}$$

whereas the PWA virtual sensor (10.1) is based on the PWA model

$$\Sigma_{Pf}: \left\{\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}_i\boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) \quad \text{for } \boldsymbol{x}_f(t) \in \Lambda_i,\ i \in \{1,\dots,r\}. \right.$$

The difference between the input-affine model and the PWA model,

$$\boldsymbol{\varepsilon}(\boldsymbol{x}_f) = \boldsymbol{f}(\boldsymbol{x}_f) - \boldsymbol{A}_i\boldsymbol{x}_f - \boldsymbol{a}_{f,i} + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d} \quad \text{for } \boldsymbol{x}_f \in \Lambda_i,\ i \in \{1,\dots,r\}, \tag{10.16}$$

consists of two parts: the term $\boldsymbol{f}(\boldsymbol{x}_f) - \boldsymbol{A}_i\boldsymbol{x}_f - \boldsymbol{a}_{f,i}$ represents the nonlinear approximation error, whereas the term $\boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}$ represents the omitted disturbance influence, since the disturbance is generally not available for measurement. Using the model error (10.16), the input-affine system (10.15) is re-written as a perturbed PWA model

$$\Sigma_{Pf,NL}: \left\{\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}_i\boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{\varepsilon}(\boldsymbol{x}_f(t)) \quad \text{for } \boldsymbol{x}_f(t) \in \Lambda_i,\ i \in \{1,\dots,r\}. \right.$$

Suppose that the model approximation error is globally bounded for vanishing disturbance:

$$\forall \boldsymbol{x}_f \in \mathbb{R}^n: \ \|\boldsymbol{\varepsilon}(\boldsymbol{x}_f)\| \leq E \quad \text{for } \boldsymbol{d} \equiv \boldsymbol{0}.$$

This assumption is usually satisfied, and input-affine systems with constant input gain can be approximated by the class of PWA models (9.1) to arbitrary precision

[90]. Including the modelling error into the observation error (10.5) leads to the new dynamics for the observation error and the difference system

$$\Sigma_e : \begin{cases} \dot{\boldsymbol{e}}(t) &= \boldsymbol{k}_e(\boldsymbol{x}_f(t)+\boldsymbol{e}(t)) - \boldsymbol{k}_e(\boldsymbol{x}_f(t)) + \begin{pmatrix}\boldsymbol{B_d} & \boldsymbol{I}\end{pmatrix}\begin{pmatrix}\boldsymbol{d}(t) \\ \boldsymbol{\varepsilon}(\boldsymbol{x}_f(t))\end{pmatrix} \\ & \text{where } \boldsymbol{k}_e(\boldsymbol{\xi}) \triangleq (\boldsymbol{A}_i - \boldsymbol{L}\boldsymbol{C}_f)\boldsymbol{\xi} + \boldsymbol{a}_{f,i},\ \boldsymbol{\xi} \in \Lambda_i,\ i \in \{1,\dots,r\}, \\ \Sigma_\Delta : \begin{cases} \dot{\boldsymbol{x}}_\Delta(t) &= \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t)) - \boldsymbol{k}_\Delta(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t)) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t) + \boldsymbol{B}\boldsymbol{u}_c(t) \\ & \text{where } \boldsymbol{k}_\Delta(\boldsymbol{\eta}) \triangleq (\boldsymbol{A}_j - \boldsymbol{B}_f\boldsymbol{M})\boldsymbol{\eta} + \boldsymbol{a}_{f,j},\ \boldsymbol{\eta} \in \Lambda_j,\ j \in \{1,\dots,r\}. \end{cases} \end{cases} \tag{10.17}$$

Note that the difference system $\Sigma_\Delta$ does not change. The only influence of the modelling error is on the observation error system $\Sigma_{\boldsymbol{e}}$. The following result is obtained.

**Theorem 10.2 (Robust PWA virtual sensor and PWA virtual actuator).** *The observation error (10.17) is ISpS w.r.t. the disturbance* $\boldsymbol{d}$*. The reconfigured closed-loop system* $(\Sigma_{Pf,NL}, \Sigma_S, \Sigma_A, \Sigma_C)$ *is ISpS with respect to the input* $(\boldsymbol{r}, \boldsymbol{d})$*.*

*Proof.* See Appendix D, page 267.

The ISpS property still implies that all system states remain bounded for bounded reference and disturbance inputs. However, for zero references and zero disturbance, the system states do not converge to the origin any longer, but to a spherical neighborhood of the origin whose radius depends on the magnitude of the model uncertainties. If additional well-behavedness of solutions inside that ball were needed, then an supplementary small-gain condition would be needed for the feedback interconnection of the observation error and the model uncertainties.

This result implies that the reconfiguration scheme based on PWA system models may be used with nonlinear systems approximated by the PWA model. The model uncertainties cause variation of the state whose bound is proportional to the model approximation error. Due to the observability of the observation error from the output $\boldsymbol{y}_c$, the controller may reject the disturbance induced by the modelling error. This is shown by means of the example in the next chapter, where the recovery of closed-loop tracking is pursued.

## 10.5 Duality between Piecewise Affine Virtual Sensor and Piecewise Affine Virtual Actuator

The PWA virtual sensor $\Sigma_S$ and the PWA virtual actuator $\Sigma_A$ are dual systems in the following sense.

**Theorem 10.3 (Duality between the PWA virtual sensor and the PWA virtual actuator).** *Any solution $\boldsymbol{L}$ to the PWA virtual sensor design problem for the pair $(\boldsymbol{C}_f, \boldsymbol{A}_i)$ also parameterises a corresponding solution $\boldsymbol{M}$ to the PWA virtual actuator design problem for the pair $(\boldsymbol{A}_i, \boldsymbol{B}_f) = (\boldsymbol{A}_i^T, \boldsymbol{C}_f^T)$. The solutions $\boldsymbol{L}$ and $\boldsymbol{M}$ are linked by means of the relation $\boldsymbol{L} = \boldsymbol{M}^T$.*

*Proof.* See Appendix D, page 268.

This result implies that the duality property linking the linear virtual sensor to the linear virtual actuator is also true in the case of PWA systems. The result holds in the sense that an approach for obtaining suitable gains for one system can be used to obtain suitable gains of the other system.

## 10.6 Summary and Discussion

This chapter has provided a solution to the stability recovery problem after actuator and sensor faults in PWA systems, where the faults are modelled as changed input and output matrices and a changed affine term. The fault model can represent actuators that are blocked in arbitrary positions. The solution consists of a PWA virtual sensor and a PWA virtual actuator that are generalisations of their linear counterparts. As in the linear case, the fault-hiding property is a consequence of the internal structures of the PWA virtual sensor and the PWA virtual actuator (Lemma 10.1). Further properties, such as stability, depend on the choice of their free parameters.

The main result is a set of sufficient conditions for reconfigured closed-loop ISS given in LMI form (Theorem 10.1). A necessary and sufficient condition (9.9) for the possibility of compensating stuck actuators was previously given. The conditions are also used to formulate a procedure for determining the gains of the PWA virtual sensor and the PWA virtual actuator (Algorithm 10.1). Although the conditions are sufficient but not necessary, they provide useful solutions in numerous practical cases. With respect to the requirements stated in Chapter 1.3, the method is suitable for autonomous online application. Given the current computational power, it is numerically applicable to small- and medium-size problems. Furthermore, the method is robust against uncertainties of the PWA model approximating the plant dynamics (Theorem 10.2). The duality found between the design procedures of the linear virtual sensor and the linear virtual actuator extends to their PWA counterparts (Theorem 10.3).

The following chapter extends the approach taken in this chapter from stability recovery to tracking recovery.

# Chapter 11
# Setpoint Tracking Recovery after Actuator and Sensor Faults in Piecewise Affine Systems

**Abstract.** This chapter extends the stability recovery approach to the reconfigurable control of piecewise affine systems towards the tracking recovery for constant reference inputs in the presence of constant disturbances. The extensions are based on internal models of the reference and disturbance signals and on the convergence property. The reconfiguration scheme is shown to be robust against uncertainties in the piecewise affine model of the faulty plant, against time-varying disturbances, and against uncertainties in the fault diagnosis result.

## 11.1 Rejection of Measured Disturbances

In this chapter, Problem 9.2 is solved. In other words, the question is addressed how to ensure that the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_S, \Sigma_A, \Sigma_C)$ tracks constant reference inputs $\boldsymbol{r}$ in the presence of constant disturbances with stable dynamics. It is assumed for the remainder of this chapter that the nominal closed-loop system $(\Sigma_P, \Sigma_C)$ tracks constant reference signals in the sense that Assumption 9.3 will be a standing assumption.

This section addresses the comparatively easy case where the disturbance signal $\boldsymbol{d}$ acting on the faulty system (9.7) is measured. Otherwise, internal-model-based extensions of the PWA virtual sensor and the PWA virtual actuator are used as shown in Section 11.2 below.

The disturbance measurement is used in the PWA virtual sensor (10.1), which changes to the augmented PWA virtual sensor.

**Definition 11.1 (Augmented piecewise affine virtual sensor).** The *augmented piecewise affine virtual sensor* is the dynamical system

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 169–189.
springerlink.com

$$\Sigma_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= (\boldsymbol{A}_i - \boldsymbol{L}\boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{L}\boldsymbol{y}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \\ & \quad \text{for } \hat{\boldsymbol{x}}_f \in \Lambda_i,\ i \in \{1,\dots,r\} \\ \hat{\boldsymbol{y}}_c(t) &= \boldsymbol{y}_f(t) + (\boldsymbol{C} - \boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) \end{cases} \tag{11.1}$$

with the initial condition $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$. ◇

Consequently, the dynamics (10.5), (10.6) of the observation error changes to

$$\Sigma_e : \left\{ \dot{\boldsymbol{e}}(t) = \boldsymbol{k}_e(\boldsymbol{x}_f(t) + \boldsymbol{e}(t)) - \boldsymbol{k}_e(\boldsymbol{x}_f(t)), \right. \tag{11.2}$$

where the function $\boldsymbol{k}_e(\cdot)$ is defined in Equation (10.6). Hence, the disturbance is not an input to the error subsystem $\Sigma_e$ any longer, and the state estimate is globally asymptotically stable for arbitrary disturbances. Since the disturbance is measurable, typically the nominal controller (9.6) uses the disturbance measurement, i.e. $\boldsymbol{u}_c(t) = \Omega_C(\boldsymbol{r}, \boldsymbol{y}, \boldsymbol{d}, \boldsymbol{x}_{c0})$. It is also still assumed that the modified nominal closed-loop system satisfies Assumption 9.2. The $\Sigma_{\tilde{P}}$-subsystem of the reconfigured plant as written in Equation (10.2), to which the nominal controller $\Sigma_C$ is connected, does, however, not depend on the disturbance. Therefore, the reference model (10.2) is augmented by the disturbance input, leading to the augmented PWA virtual actuator.

**Definition 11.2 (Augmented piecewise affine virtual actuator).** The *augmented piecewise affine virtual actuator* is the dynamical system

$$\Sigma_A : \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) &= \boldsymbol{A}_j\tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) \text{ for } \tilde{\boldsymbol{x}} \in \Lambda_j,\ j \in \{1,\dots,r\} \\ \boldsymbol{y}_c(t) &= \hat{\boldsymbol{y}}_c(t) + \boldsymbol{C}\boldsymbol{x}_{\Delta}(t) \\ \boldsymbol{u}_f(t) &= \boldsymbol{M}\boldsymbol{x}_{\Delta}(t) + \boldsymbol{B}_f^{+}\boldsymbol{a}_{\Delta,j} \end{cases} \tag{11.3}$$

with the initial condition $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_{f0}$, and where $\boldsymbol{x}_{\Delta}(t) = \tilde{\boldsymbol{x}}(t) - \hat{\boldsymbol{x}}_f(t)$ in accordance with Equation (3.35). ◇

The modified reconfiguration block $\Sigma_R = (\Sigma_S, \Sigma_A)$ defined by the equations (11.2) and (11.3) is now used in the closed-loop system. In this case, the following result is in place.

**Lemma 11.1 (Measured disturbance compensation).** *Consider the faulty PWA system (9.7), suppose that the disturbance $\boldsymbol{d}$ is measurable, and suppose that Assumption 9.1 and Assumption 9.4 hold. If there exist matrices $\boldsymbol{X}_s \in \mathbb{R}^{n\times n}$, $\boldsymbol{X}_s = \boldsymbol{X}_s^T \succ 0$ and $\boldsymbol{Y}_s \in \mathbb{R}^{n\times q}$ that satisfy the LMIs (10.11), (10.13), then the system (11.1) with the gain $\boldsymbol{L} = \boldsymbol{X}_s^{-1}\boldsymbol{Y}_s$ is an observer for the system (9.7) with globally exponentially*

*stable observation error dynamics (11.2), thus all solutions* $\boldsymbol{e}(t)$ *of the observation error (11.2) satisfy the relation*

$$\|\boldsymbol{e}(t)\| \leq ce^{-at}\|\boldsymbol{e}(0)\|,\ t \in [0,\infty), \tag{11.4}$$

*where the real numbers* $c > 0$ *and* $a > 0$ *depend only on the solutions of the LMIs (10.11), (10.13). The reconfiguration block (11.1), (11.3) satisfies the weak fault-hiding goal as well as Equation (11.17) for arbitrary disturbances.*

*Proof.* The fault-hiding aspect is immediate from Assumption 9.3 and the proof of Lemma 10.1. The bound (11.4) follows immediately from the observation error dynamics (11.2) and similar reasoning as in the proof of Lemma 10.2. ■

In applications, the disturbance is frequently not measured, and furthermore, the tracking of constant reference inputs has not been considered yet. These aspects are addressed in the following section.

## 11.2 Extended Piecewise Affine Virtual Sensor and Extended Piecewise Affine Virtual Actuator

This section investigates the case of unmeasured but constant disturbances. The idea consists in the estimation of the disturbance by means of a disturbance observer. The constant disturbance is assumed to be generated by a exogenous system defined by the equation

$$\dot{\boldsymbol{d}}(t) = \boldsymbol{0},\ \boldsymbol{d}(0) = \boldsymbol{d}_0$$

with unknown initial condition $\boldsymbol{d}_0$. The problem consists in obtaining an estimate $\hat{\boldsymbol{d}}$ of the disturbance $\boldsymbol{d}$. The estimate $\hat{\boldsymbol{d}}$ is then used as an input to the PWA virtual sensor. This idea leads to the extended PWA virtual sensor.

**Definition 11.3 (Extended piecewise affine virtual sensor).** The *extended piecewise affine virtual sensor* is the dynamical system

$$\bar{\Sigma}_S : \begin{cases} \dot{\hat{\boldsymbol{x}}}_f(t) &= (\boldsymbol{A}_i - \boldsymbol{L}\boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f\boldsymbol{u}_f(t) + \boldsymbol{L}\boldsymbol{y}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\hat{\boldsymbol{d}}(t) \\ & \quad \text{for } \hat{\boldsymbol{x}}_f \in \Lambda_i,\ i \in \{1,\ldots,r\} \\ \dot{\hat{\boldsymbol{d}}}(t) &= \boldsymbol{L}_d\left(\boldsymbol{y}_f(t) - \boldsymbol{C}_f\hat{\boldsymbol{x}}_f(t)\right) \\ \hat{\boldsymbol{y}}_c(t) &= \boldsymbol{P}\boldsymbol{y}_f(t) + (\boldsymbol{C} - \boldsymbol{P}\boldsymbol{C}_f)\hat{\boldsymbol{x}}_f(t) \end{cases} \tag{11.5}$$

with the initial conditions $\hat{\boldsymbol{x}}_f(0) = \hat{\boldsymbol{x}}_{f0}$ and $\hat{\boldsymbol{d}}(0) = \hat{\boldsymbol{d}}_0$. ◇

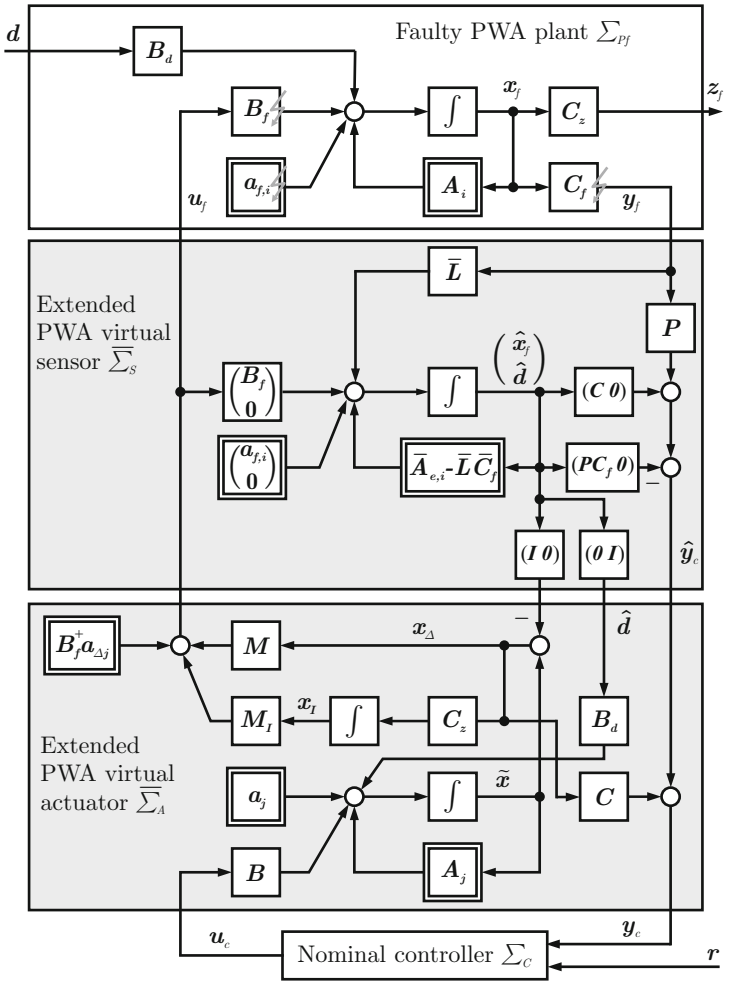


**Fig. 11.1** Extended PWA virtual sensor and extended PWA virtual actuator for tracking recovery.

The extended PWA virtual sensor that replaces the PWA virtual sensor (10.1) is shown in Fig. 11.1. In addition to the state observation error $\boldsymbol{e}$ defined in Equation (3.39), the disturbance observation error $\boldsymbol{e}_d$ is defined as

$$\boldsymbol{e}_d(t) = \hat{\boldsymbol{d}}(t) - \boldsymbol{d}(t),\ \boldsymbol{e}_d(0) = \hat{\boldsymbol{d}}_0 - \boldsymbol{d}_0. \tag{11.6}$$

The disturbance estimation error has an initial value that is defined by the disturbance observer initial value $\hat{\boldsymbol{d}}_0$ and by the true initial disturbance value $\boldsymbol{d}_0$. The gains $\boldsymbol{L}$ and $\boldsymbol{L}_d$ are designed to stabilise the extended system (11.5), which is rewritten in terms of the extended observation error $\bar{\boldsymbol{e}}$ and the extended state $\bar{\boldsymbol{x}}$

$$\bar{\boldsymbol{e}}(t) = \begin{pmatrix} \boldsymbol{e}(t) \\ \boldsymbol{e}_d(t) \end{pmatrix}, \; \bar{\boldsymbol{x}}(t) = \begin{pmatrix} \boldsymbol{x}_f(t) \\ \boldsymbol{d}(t) \end{pmatrix} \tag{11.7}$$

as

$$\bar{\Sigma}_e : \left\{ \dot{\bar{\boldsymbol{e}}}(t) \;\; = \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{e}}(t) + \bar{\boldsymbol{x}}(t)) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t)), \right. \tag{11.8}$$

where $\bar{\boldsymbol{e}}(0) = \bar{\boldsymbol{e}}_0 = (\boldsymbol{e}(0)^T \; \boldsymbol{e}_d(0))^T$ and

$$\bar{\boldsymbol{k}}_e\left(\begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix}\right) \triangleq (\bar{\boldsymbol{A}}_{e,i} - \bar{\boldsymbol{L}}\bar{\boldsymbol{C}}_f)\begin{pmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{pmatrix} + \bar{\boldsymbol{a}}_{f,i} \text{ for } \boldsymbol{\xi} \in \Lambda_i, \; i \in \{1, \ldots, r\} \tag{11.9}$$

$$\bar{\boldsymbol{A}}_{e,i} \triangleq \begin{pmatrix} \boldsymbol{A}_i & \boldsymbol{B}_d \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}, \; \bar{\boldsymbol{a}}_{f,i} \triangleq \begin{pmatrix} \boldsymbol{a}_{f,i} \\ \boldsymbol{0} \end{pmatrix}, \; \bar{\boldsymbol{L}} \triangleq \begin{pmatrix} \boldsymbol{L} \\ \boldsymbol{L}_d \end{pmatrix}, \; \bar{\boldsymbol{C}}_f \triangleq \begin{pmatrix} \boldsymbol{C}_f & \boldsymbol{0} \end{pmatrix}. \tag{11.10}$$

The free parameters $\boldsymbol{L}$ and $\boldsymbol{L}_d$ will be used to stabilise the extended observation error $\bar{\boldsymbol{e}}$. The free matrix parameter $\boldsymbol{P}$ has no effect on stability and may be used, for example, to pass through healthy measurements. Note that the disturbance $\boldsymbol{d}$ is not a genuine input to (11.8). Next, the nulling of the output term $\boldsymbol{C}_z\boldsymbol{x}_\Delta(t)$ is addressed, which was identified as the second part (11.18) of the translated problem to be solved for achieving tracking stated in Lemma 11.3. It is assumed that either the disturbance is measured as described in Section 11.1, or an extended PWA virtual sensor (11.5) that also estimates constant disturbances is available so that the first subproblem has been solved.

To address the reference tracking aspect, it is assumed that the reference signal is generated by an exogenous system

$$\dot{\boldsymbol{r}}(t) = \boldsymbol{0}, \quad \boldsymbol{r}(0) = \boldsymbol{r}_0.$$

The PWA virtual actuator is augmented by an internal model of the constant reference signal. In other words, integrators with states $\boldsymbol{x}_I \in \mathbb{R}^p$ are added. The number of integrators is chosen to match the number of components of the output $\boldsymbol{z}$. The PWA virtual actuator (10.2) is hence replaced by the extended PWA virtual actuator defined as follows.

**Definition 11.4 (Extended piecewise affine virtual actuator).** The *extended piecewise affine virtual actuator* is the dynamical system

$$\bar{\Sigma}_A : \begin{cases} \dot{\tilde{\boldsymbol{x}}}(t) & = \boldsymbol{A}_j\tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B}_{\boldsymbol{d}}\hat{\boldsymbol{d}}(t) \text{ for } \tilde{\boldsymbol{x}} \in \Lambda_j, \; j \in \{1, \ldots, r\} \\ \dot{\boldsymbol{x}}_I(t) & = \boldsymbol{C}_z\boldsymbol{x}_\Delta(t) \\ \boldsymbol{y}_c(t) & = \hat{\boldsymbol{y}}_c(t) + \boldsymbol{C}\boldsymbol{x}_\Delta(t) \\ \boldsymbol{u}_f(t) & = \boldsymbol{M}\boldsymbol{x}_\Delta(t) + \boldsymbol{M}_I\boldsymbol{x}_I(t) + \boldsymbol{B}_f^+\boldsymbol{a}_{\Delta,j} \end{cases} \tag{11.11}$$

with the initial conditions $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_{f0}$ and $\boldsymbol{x}_I(0) = \boldsymbol{0}$, and where $\boldsymbol{x}_\Delta(t) = \tilde{\boldsymbol{x}}(t) - \hat{\boldsymbol{x}}_f(t)$ in accordance with Equation (3.35). ◇

The extended PWA virtual actuator is shown in Fig. 11.1. Using the function

$$\bar{\boldsymbol{k}}_{\Delta}\left(\begin{pmatrix}\boldsymbol{\xi}\\ \boldsymbol{\eta}\end{pmatrix}\right) \triangleq \left(\bar{\boldsymbol{A}}_{a,j} - \bar{\boldsymbol{B}}_f \bar{\boldsymbol{M}}\right)\begin{pmatrix}\boldsymbol{\xi}\\ \boldsymbol{\eta}\end{pmatrix} + \bar{\boldsymbol{a}}_{f,j} \text{ for } \boldsymbol{\xi} \in \Lambda_j,\ j \in \{1,\dots,r\}, \tag{11.12}$$

where

$$\bar{\boldsymbol{A}}_{a,j} \triangleq \begin{pmatrix}\boldsymbol{A}_j & \boldsymbol{0}\\ \boldsymbol{C}_z & \boldsymbol{0}\end{pmatrix},\ \bar{\boldsymbol{a}}_{f,j} \triangleq \begin{pmatrix}\boldsymbol{a}_{f,j}\\ \boldsymbol{0}\end{pmatrix},\ \bar{\boldsymbol{B}}_f \triangleq \begin{pmatrix}\boldsymbol{B}_f\\ \boldsymbol{0}\end{pmatrix},\ \bar{\boldsymbol{M}} \triangleq \begin{pmatrix}\boldsymbol{M} & \boldsymbol{M}_I\end{pmatrix}, \tag{11.13}$$

and using Assumption 9.5, it is straightforward to obtain the following combined dynamics of the extended difference system

$$\bar{\Sigma}_{\Delta}: \left\{\begin{pmatrix}\dot{\boldsymbol{x}}_{\Delta}(t)\\ \dot{\boldsymbol{x}}_I(t)\end{pmatrix} = \bar{\boldsymbol{k}}_{\Delta}\left(\begin{pmatrix}\tilde{\boldsymbol{x}}(t)\\ \boldsymbol{x}_I(t)\end{pmatrix}\right) - \bar{\boldsymbol{k}}_{\Delta}\left(\begin{pmatrix}\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)\\ \boldsymbol{0}\end{pmatrix}\right) + \begin{pmatrix}\boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t)\\ \boldsymbol{0}\end{pmatrix}. \tag{11.14}$$

The following Lemma shows that the reconfigured plant obtained with extended virtual sensor $\bar{\Sigma}_S$ and the extended virtual actuator $\bar{\Sigma}_A$ satisfies the weak fault-hiding goal.

**Lemma 11.2 (Weak fault-hiding).** *The reconfigured plant $\Sigma_{Pr} = (\Sigma_{Pf}, \bar{\Sigma}_S, \bar{\Sigma}_A)$ formed by the faulty PWA plant (9.7), the extended PWA virtual sensor (11.5), and the extended PWA virtual actuator (11.11) satisfies the weak fault-hiding goal.*

*Proof.* Based on Assumption 9.5, the relevant part of the reconfigured plant model is given by the equations (index $j$ determined by $\tilde{\boldsymbol{x}}(t) \in \Lambda_j$)

$$\begin{pmatrix}\dot{\tilde{\boldsymbol{x}}}(t)\\ \dot{\bar{\boldsymbol{e}}}(t)\\ \dot{\boldsymbol{x}}_{\Delta}(t)\\ \dot{\boldsymbol{x}}_I(t)\end{pmatrix} = \begin{pmatrix}\boldsymbol{A}_j\tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j + \boldsymbol{B}_d\hat{\boldsymbol{d}}(t)\\ \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{e}}(t) + \bar{\boldsymbol{x}}(t)) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t))\\ \boldsymbol{k}_{\Delta}(\tilde{\boldsymbol{x}}(t)) - \boldsymbol{k}_{\Delta}(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t)) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t)\\ \boldsymbol{C}_z\boldsymbol{x}_{\Delta}(t)\end{pmatrix} + \begin{pmatrix}\boldsymbol{B}\\ \boldsymbol{0}\\ \boldsymbol{B}\\ \boldsymbol{0}\end{pmatrix}\boldsymbol{u}_c(t),$$

$$\boldsymbol{y}_c(t) = \begin{pmatrix}\boldsymbol{C} & -\boldsymbol{P}\bar{\boldsymbol{C}}_f & \boldsymbol{0} & \boldsymbol{0}\end{pmatrix}\begin{pmatrix}\tilde{\boldsymbol{x}}(t)\\ \bar{\boldsymbol{e}}(t)\\ \boldsymbol{x}_{\Delta}(t)\\ \boldsymbol{x}_I(t)\end{pmatrix},\ \begin{pmatrix}\tilde{\boldsymbol{x}}(0)\\ \bar{\boldsymbol{e}}(0)\\ \boldsymbol{x}_{\Delta}(0)\\ \boldsymbol{x}_I(0)\end{pmatrix} = \begin{pmatrix}\hat{\boldsymbol{x}}_{f,0}\\ \bar{\boldsymbol{e}}_0\\ \boldsymbol{0}\\ \boldsymbol{0}\end{pmatrix},$$

where $\boldsymbol{k}_{\Delta}(\boldsymbol{\xi}) \triangleq (\boldsymbol{A}_j - \boldsymbol{B}_f\boldsymbol{M})\boldsymbol{\xi} + \boldsymbol{a}_{f,j}$ for $\boldsymbol{\xi} \in \Lambda_j, j \in \{1,\dots,r\}$. Weak fault hiding (Definition 3.6) is achieved by the matching initialisation $\hat{\boldsymbol{x}}_{f,0} = \boldsymbol{x}_0$, $\hat{\boldsymbol{d}}_0 = \boldsymbol{0}$ which implies that $\bar{\boldsymbol{e}}_0 = \boldsymbol{0}$ for $\boldsymbol{d}(t) = \boldsymbol{0}\ \forall t \in \mathbb{R}_+$, and $\dot{\hat{\boldsymbol{d}}}(t) \equiv \boldsymbol{0}$. The nominal controller is attached to the reference system

$$\Sigma_{\tilde{P}}: \left\{\dot{\tilde{\boldsymbol{x}}}(t) = \boldsymbol{A}_j\tilde{\boldsymbol{x}}(t) + \boldsymbol{a}_j + \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{B}_d\hat{\boldsymbol{d}}(t) \text{ for } \tilde{\boldsymbol{x}} \in \Lambda_j, j \in \{1,\dots,r\}\right.$$

governed by nominal dynamics. The reference state $\tilde{\boldsymbol{x}}$ is decoupled from the observation error $\bar{\boldsymbol{e}}$ and the difference state $\boldsymbol{x}_{\Delta}$ for $\hat{\boldsymbol{d}}_0 = \boldsymbol{0}$ since $\boldsymbol{d}_0 = \boldsymbol{0}$. The output $\boldsymbol{y}_c$ depends on $\tilde{\boldsymbol{x}}$ and $\bar{\boldsymbol{e}}$, where the observation error $\bar{\boldsymbol{e}}$ is autonomous, and $\boldsymbol{d} \equiv \boldsymbol{0}$ and $\hat{\boldsymbol{d}}(0) = \boldsymbol{0}$ imply that $\bar{\boldsymbol{e}} \equiv \boldsymbol{0}$. ∎

The next section reveals how the setpoint tracking recovery problem relates to the synthesis of the free parameters of the extended PWA virtual sensor and the extended PWA virtual actuator. Finally, a procedure is shown that ensures that the disturbance estimate exponentially converges to the true disturbance, and that the output of the faulty plant recovers the nominal tracking precision.

## 11.3 Transformation of the Problem

The relevant output $\boldsymbol{z}$ is defined in Equation (9.1). It is studied in this section how and under which conditions the extensions made in the previous section guarantee that the corresponding output $\boldsymbol{z}_f$ of the faulty system defined in Equation (9.7) asymptotically tracks the reference input to the same precision $K \geq 0$ as in the nominal case:

$$\limsup_{t\to\infty} \|\boldsymbol{e}_z(t)\| \triangleq \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{z}_f(t)\| = \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{C}_z \boldsymbol{x}_f(t)\| \leq K. \tag{11.15}$$

From the definitions (3.35) and (3.39) of the difference state $\boldsymbol{x}_\Delta$ and the observation error $\boldsymbol{e}$, one obtains the equivalent goal

$$\limsup_{t\to\infty} \|\boldsymbol{e}_z(t)\| = \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{C}_z \tilde{\boldsymbol{x}}(t) + \boldsymbol{C}_z(\boldsymbol{x}_\Delta(t) + \boldsymbol{e}(t))\| \leq K, \tag{11.16}$$

where it is known from Assumption 9.3 that $\limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{C}_z \tilde{\boldsymbol{x}}(t)\| \leq K$, if the state $\tilde{\boldsymbol{x}}$ is governed by nominal dynamics. In the previous chapter, it has turned out that the observation error $\boldsymbol{e}$ acts on the transformed closed-loop system as a measurement disturbance entering at the output $\boldsymbol{y}_c$ (see the proof of Theorem 10.1). It is, therefore, sufficient for solving Problem 9.2 that $\lim_{t\to\infty} \boldsymbol{C}_z(\boldsymbol{e}(t) + \boldsymbol{x}_\Delta(t)) = \boldsymbol{0}$ and $\lim_{t\to\infty} \bar{\boldsymbol{e}}(t) = \boldsymbol{0}$. Observing that $\bar{\boldsymbol{e}}$ and $\boldsymbol{x}_\Delta$ are driven by different exogenous inputs $\boldsymbol{d}$ and $\boldsymbol{u}_c$ of which the first one is generally unknown, achieving the special case $\boldsymbol{C}_z\boldsymbol{e}(\infty) = -\boldsymbol{C}_z\boldsymbol{x}_\Delta(\infty)$ is unrealistic in most cases. Therefore, the output $\boldsymbol{C}_z\boldsymbol{e}$ will be decoupled from $\boldsymbol{d}$, and $\boldsymbol{C}_z\boldsymbol{x}_\Delta$ will be decoupled from $\boldsymbol{u}_c$, which leads to the following sufficient conditions for solving Problem 9.2.

**Lemma 11.3 (Setpoint tracking recovery for PWA systems).** *The reconfigured closed-loop system recovers the nominal closed-loop tracking precision $K$ if the reconfiguration block $\Sigma_R$ guarantees that the following relations hold for constant reference inputs and constant disturbance inputs:*

$$\lim_{t\to\infty} \bar{\boldsymbol{e}}(t) = \boldsymbol{0} \tag{11.17}$$

$$\lim_{t\to\infty} \boldsymbol{C}_z \boldsymbol{x}_\Delta(t) = \boldsymbol{0}. \tag{11.18}$$

For fault scenarios with blocked actuators, tracking will only be achievable if the effect of blocked actuators can be compensated. Therefore, Assumption 9.5 still holds.

The first step consists in asymptotically decoupling the observation error (11.17) from the disturbance (the case of measurable disturbance signals has already been discussed in Section 11.1). The second step consists in decoupling the output-relevant difference system (11.18) from the control input. The main result discussed in Section 11.4 consists in at set of sufficient conditions imposed on the parameters of the extended PWA virtual actuator and extended PWA virtual sensor that guarantee both the reconfigured closed-loop ISS property and the tracking of constant setpoints in the presence of constant disturbances. The reconfiguration scheme is robust against inaccurate models of the faulty plant and time-varying disturbances, as shown in Section 11.5. Furthermore, robustness against uncertainties in the fault diagnosis result is shown in Section 11.6. The extended PWA virtual sensor and the extended PWA virtual actuator are dual systems as shown in Section 11.7. The chapter is summarised in Section 11.8.

## 11.4 Main Stability and Tracking Result

This section provides the solution to Problem 9.2 based on the extended PWA virtual sensor $\bar{\Sigma}_S$ and the extended PWA virtual actuator $\bar{\Sigma}_A$ defined in the previous section.

**Lemma 11.4 (Global exponential stability of the extended observation error).** *Consider the faulty PWA system (9.7), suppose that Assumptions 9.1 and 9.4 hold, and assume that the disturbance is constant ($\dot{\boldsymbol{d}}(t) = \boldsymbol{0}$). If there exist matrices $\bar{\boldsymbol{X}}_s \in \mathbb{R}^{(n+k)\times(n+k)}$, $\bar{\boldsymbol{X}}_s = \bar{\boldsymbol{X}}_s^T \succ 0$, and $\bar{\boldsymbol{Y}}_s \in \mathbb{R}^{(n+k)\times(q+k)}$ that satisfy the LMIs*

$$\bar{\boldsymbol{X}}_s \bar{\boldsymbol{A}}_{e,i} + \bar{\boldsymbol{A}}_{e,i}^T \bar{\boldsymbol{X}}_s - \bar{\boldsymbol{Y}}_s \bar{\boldsymbol{C}}_f - \bar{\boldsymbol{C}}_f^T \bar{\boldsymbol{Y}}_s^T \prec 0,\ i = 1,\ldots,r, \tag{11.19}$$

*then the system (11.5) with the gain $\bar{\boldsymbol{L}} \triangleq \bar{\boldsymbol{X}}_s^{-1} \bar{\boldsymbol{Y}}_s$ is a state and disturbance observer for the faulty system (9.7). All solutions $\bar{\boldsymbol{e}}$ of the extended observation error system defined in Equation (11.7) satisfy the relation*

$$\|\bar{\boldsymbol{e}}(t)\| \leq c e^{-at} \|\bar{\boldsymbol{e}}(0)\|,\ t \in [0, \infty), \tag{11.20}$$

*where the real numbers $c > 0$ and $a > 0$ depend only on $\bar{\boldsymbol{X}}_s$ and $\bar{\boldsymbol{Y}}_s$.*

*Proof.* Noting that the function (11.9) is continuous, a Lyapunov function $V(\bar{\boldsymbol{e}}) = \frac{1}{2}\bar{\boldsymbol{e}}^T \boldsymbol{X} \bar{\boldsymbol{e}}$ is constructed. Theorem 9.1 implies that

$$\dot{V}(\bar{\boldsymbol{e}}) = \bar{\boldsymbol{e}}^T \boldsymbol{X} \dot{\bar{\boldsymbol{e}}} = \bar{\boldsymbol{e}}^T \boldsymbol{X} (\bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}} + \bar{\boldsymbol{e}}) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}})) \tag{11.21}$$

$$\leq -a \bar{\boldsymbol{e}}^T \boldsymbol{X} \bar{\boldsymbol{e}},\ a > 0, \tag{11.22}$$

which immediately implies the inequality (11.20). The inequality (11.22) follows from Theorem 9.1 if the extended system satisfies

$$\tilde{\boldsymbol{X}}(\bar{\boldsymbol{A}}_{e,i} - \bar{\boldsymbol{L}}\bar{\boldsymbol{C}}_f) + (\bar{\boldsymbol{A}}_{e,i} - \bar{\boldsymbol{L}}\bar{\boldsymbol{C}}_f)^T \tilde{\boldsymbol{X}} \prec 0,\ i = 1,\ldots,r,$$

which are equivalent to the LMIs (11.19) after the linearising change of variables $\bar{\boldsymbol{X}}_s = \tilde{\boldsymbol{X}}$ and $\bar{\boldsymbol{Y}}_s = \tilde{\boldsymbol{X}}\bar{\boldsymbol{L}}$. ∎

Condition (11.19) ensures that the extended PWA virtual sensor (11.5) estimates both the constant disturbance and the system state in spite of the different discrete modes of the plant and the virtual sensor.

*Remark 11.1 (Relations to linear unknown-input observers).* The feasibility conditions (11.19) may be interpreted as follows, where the notation $\langle \ker \boldsymbol{C} | \boldsymbol{A} \rangle$ denotes the largest $\boldsymbol{A}$-invariant subspace contained in $\ker \boldsymbol{C}$, and $\sigma(\boldsymbol{A}|\mathcal{S})$ denotes the set of eigenvalues of $\boldsymbol{A}$ restricted to the $\boldsymbol{A}$-invariant subspace $\mathcal{S}$. It is certainly necessary (but not sufficient in general) for the observability of the faulty PWA system (9.7) that the pair $(\bar{\boldsymbol{C}}_f, \bar{\boldsymbol{A}}_{e,i})$ be detectable for all discrete modes $i \in \{1,\dots,r\}$. Detectability of the linear systems defined for each mode is equivalent [211] to the condition

$$\forall i \in \{1,\dots,r\}: \ \sigma\left(\begin{pmatrix} \boldsymbol{A}_i & \boldsymbol{B_d} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \middle| \left\langle \ker \begin{pmatrix} \boldsymbol{C}_f & \boldsymbol{0} \end{pmatrix} \middle| \begin{pmatrix} \boldsymbol{A}_i & \boldsymbol{B_d} \\ \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \right\rangle\right) \subset \mathbb{C}_-,$$

which means that in every discrete mode, the unobservable modes must be stable. Since the disturbance dynamics are marginally stable by assumption, it is necessary that the disturbance effects completely propagate through the output $\boldsymbol{C}_f$. This condition is equivalent to those found in the literature on unknown-input estimators [43, 80]. Of course, this condition alone is not sufficient for the stability of mixed modes in the plant and the virtual sensor and thus in the complete observation error dynamics, which is why the more restrictive sufficient conditions (11.19) were obtained. ∘

The following Lemma provides a sufficient condition for the ISS of the extended difference system.

**Lemma 11.5 (Input-to-state stability of the extended difference system).** *Consider the faulty PWA system (9.7) and suppose that Assumptions 9.1, 9.5 and 9.4 hold. If there exist matrices $\bar{\boldsymbol{X}}_a \in \mathbb{R}^{(n+p)\times(n+p)}$, $\bar{\boldsymbol{X}}_a = \bar{\boldsymbol{X}}_a^T \succ 0$, and $\bar{\boldsymbol{Y}}_a \in \mathbb{R}^{m\times(n+p)}$ that satisfy the LMIs*

$$\bar{\boldsymbol{A}}_{a,j}\bar{\boldsymbol{X}}_a + \bar{\boldsymbol{X}}_a\bar{\boldsymbol{A}}_{a,j}^T - \bar{\boldsymbol{B}}_f\bar{\boldsymbol{Y}}_a - \bar{\boldsymbol{Y}}_a^T\bar{\boldsymbol{B}}_f^T \prec 0, \ j = 1,\dots,r, \tag{11.23}$$

*then the extended difference system (11.14) of the extended virtual actuator (11.11) with the gain $\bar{\boldsymbol{M}} \triangleq \bar{\boldsymbol{Y}}_a\bar{\boldsymbol{X}}_a^{-1}$ is 0-GES for zero inputs $\boldsymbol{u}_c, \boldsymbol{e} \equiv \boldsymbol{0}$. Moreover, all solutions $(\boldsymbol{x}_\Delta(t)^T, \boldsymbol{x}_I(t)^T)^T$ of the unforced difference system (11.14) (i.e. $\boldsymbol{u}_c, \boldsymbol{e} \equiv \boldsymbol{0}$) satisfy the relation*

$$\|\boldsymbol{x}_\Delta(t)\| + \|\boldsymbol{x}_I(t)\| \leq ce^{-at}\left(\|\boldsymbol{x}_\Delta(0)\| + \|\boldsymbol{x}_I(0)\|\right), \tag{11.24}$$

*where the real numbers $c > 0$ and $a > 0$ depend only on $\bar{\boldsymbol{X}}_a$ and $\bar{\boldsymbol{Y}}_a$. In other words, the difference state $\boldsymbol{x}_\Delta$ asymptotically converges to the origin: $\lim_{t\to\infty} \boldsymbol{x}_\Delta(t) = \boldsymbol{0}$ for zero inputs. Furthermore, the extended difference system is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{e})$. If the steady-state control input $\boldsymbol{u}_c$ is constant and $\lim_{t\to\infty} \boldsymbol{e}(t) = \boldsymbol{0}$, then $\lim_{t\to\infty} \boldsymbol{C}_z\boldsymbol{x}_\Delta(t) = \boldsymbol{0}$.*

*Proof.* Noting that the function (11.12) is continuous, a Lyapunov function $V((\boldsymbol{x}_\Delta^T, \boldsymbol{x}_I^T)^T) = \frac{1}{2}\begin{pmatrix}\boldsymbol{x}_\Delta^T & \boldsymbol{x}_I^T\end{pmatrix}\bar{\boldsymbol{X}}_a^{-1}\begin{pmatrix}\boldsymbol{x}_\Delta \\ \boldsymbol{x}_I\end{pmatrix}$ is constructed, and Theorem 9.1 is used to obtain the following properties of its derivative

$$\begin{aligned}\dot{V}((\boldsymbol{x}_\Delta^T, \boldsymbol{x}_I^T)^T) &= \begin{pmatrix}\boldsymbol{x}_\Delta^T & \boldsymbol{x}_I^T\end{pmatrix}\bar{\boldsymbol{X}}_a^{-1}\left(\bar{\boldsymbol{k}}_\Delta(\tilde{\boldsymbol{x}}, \boldsymbol{x}_I) - \bar{\boldsymbol{k}}_\Delta(\tilde{\boldsymbol{x}} - \boldsymbol{x}_\Delta, \boldsymbol{0}) + \begin{pmatrix}\boldsymbol{B}\boldsymbol{u}_c + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e} \\ \boldsymbol{0}\end{pmatrix}\right) \\ &\leq -a\begin{pmatrix}\boldsymbol{x}_\Delta \\ \boldsymbol{x}_I\end{pmatrix}^T\bar{\boldsymbol{X}}_a^{-1}\begin{pmatrix}\boldsymbol{x}_\Delta \\ \boldsymbol{x}_I\end{pmatrix} + \begin{pmatrix}\boldsymbol{x}_\Delta \\ \boldsymbol{x}_I\end{pmatrix}^T\bar{\boldsymbol{X}}_a^{-1}\begin{pmatrix}\boldsymbol{B}\boldsymbol{u}_c + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e} \\ \boldsymbol{0}\end{pmatrix}\end{aligned}$$

for some $a > 0$, which is a Lyapunov characterisation of ISS with respect to the input $(\boldsymbol{u}_c, \boldsymbol{e})$. The latter inequality follows from Theorem 9.1 if the extended difference system satisfies the inequalities

$$\bar{\boldsymbol{X}}_a^{-1}(\bar{\boldsymbol{A}}_{a,j} - \bar{\boldsymbol{B}}_f\bar{\boldsymbol{M}}) + (\bar{\boldsymbol{A}}_{a,j} - \bar{\boldsymbol{B}}_f\bar{\boldsymbol{M}})^T\bar{\boldsymbol{X}}_a^{-1} \prec 0,\ j = 1, \ldots, r,$$

which are equivalent to the LMIs (11.23) after multiplication of the LMIs with $\bar{\boldsymbol{X}}_a$ from left and right and introduction of the new variable $\bar{\boldsymbol{Y}}_a = \bar{\boldsymbol{M}}\bar{\boldsymbol{X}}_a$. It is thus proven that the extended difference system (11.14) is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{e})$ if the given LMIs is satisfied.

It remains to be proven that $\lim_{t\to\infty}\boldsymbol{C}_z\boldsymbol{x}_\Delta(t) = \boldsymbol{0}$ as $\boldsymbol{u}_c$ becomes constant in steady-state. This property is proven by showing that the extended difference system (11.14) is exponentially convergent, and thus a constant steady-state input implies a constant steady-state solution for the extended difference system. Consider a candidate Lyapunov function $V$ for quadratic convergence:

$$V = \begin{pmatrix}(\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1})^T & (\boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1})^T\end{pmatrix}\boldsymbol{P}\begin{pmatrix}\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1} \\ \boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1}\end{pmatrix}$$

Along solutions of (11.14) the time derivative of $V$ satisfies

$$\begin{aligned}\dot{V} = &\begin{pmatrix}(\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1})^T & (\boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1})^T\end{pmatrix}\boldsymbol{P} \\ &\left(\bar{\boldsymbol{k}}_\Delta\left(\begin{pmatrix}\tilde{\boldsymbol{x}} \\ \boldsymbol{x}_{I,2}\end{pmatrix}\right) - \bar{\boldsymbol{k}}_\Delta\left(\begin{pmatrix}\tilde{\boldsymbol{x}} \\ \boldsymbol{x}_{I,1}\end{pmatrix}\right) - \bar{\boldsymbol{k}}_\Delta\left(\begin{pmatrix}\tilde{\boldsymbol{x}} - \boldsymbol{x}_{\Delta,2} \\ \boldsymbol{0}\end{pmatrix}\right) + \bar{\boldsymbol{k}}_\Delta\left(\begin{pmatrix}\tilde{\boldsymbol{x}} - \boldsymbol{x}_{\Delta,1} \\ \boldsymbol{0}\end{pmatrix}\right)\right)\end{aligned}$$

Using twice the fact that the function $\bar{\boldsymbol{k}}_\Delta$ satisfies the inequality (9.3), one obtains that there exists $a > 0$ such that

$$\dot{V} \leq -a\begin{pmatrix}\boldsymbol{0}^T & (\boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1})^T\end{pmatrix}\boldsymbol{P}\begin{pmatrix}\boldsymbol{0} \\ \boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1}\end{pmatrix} - a\begin{pmatrix}(\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1})^T & \boldsymbol{0}^T\end{pmatrix}\boldsymbol{P}\begin{pmatrix}\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1} \\ \boldsymbol{0}\end{pmatrix}. \tag{11.25}$$

Since $\boldsymbol{P} = \boldsymbol{P}^T \succ 0$, it follows that $\boldsymbol{P}$ has the structure

$$\boldsymbol{P} = \begin{pmatrix}\boldsymbol{P}_{11} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{P}_{22}\end{pmatrix} \succ 0$$

from the partition $\boldsymbol{P} = \begin{pmatrix} \boldsymbol{P}_{11} & \boldsymbol{P}_{12} \\ \boldsymbol{P}_{12} & \boldsymbol{P}_{2}2 \end{pmatrix}$. Hence, Equation (11.25) gives

$$\dot{V} \leq -a\begin{pmatrix} (\boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1})^T & (\boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1})^T \end{pmatrix} \boldsymbol{P} \begin{pmatrix} \boldsymbol{x}_{\Delta,2} - \boldsymbol{x}_{\Delta,1} \\ \boldsymbol{x}_{I,2} - \boldsymbol{x}_{I,1} \end{pmatrix}.$$

Therefore, the extended difference system (11.14) is exponentially convergent and its solutions converge to a unique constant solution, see [159, Property 2.23]. From Lemma 11.4, $\lim_{t\to\infty} \bar{\boldsymbol{e}} = \boldsymbol{0}$ for $\dot{\boldsymbol{d}} = \boldsymbol{0}$. From Assumption 9.3, it follows that $\lim_{t\to\infty} \boldsymbol{u}_c(t) = \bar{\boldsymbol{u}}_c$, so $\boldsymbol{u}_c$ is constant in steady-state and $(\boldsymbol{x}_\Delta, \boldsymbol{x}_I)$ converges to a constant steady-state solution due to [159, Property 2.25]. According to (11.11), $\boldsymbol{x}_I$ constant implies that $\boldsymbol{C}_z \boldsymbol{x}_\Delta = \boldsymbol{0}$, and it follows that $\lim_{t\to\infty} \boldsymbol{C}_z \boldsymbol{x}_\Delta(t) = \boldsymbol{0}$, as claimed. The boundedness of solutions of the extended difference system in case of non-constant inputs is guaranteed by its ISS property. ■

With the partial results regarding the extended observation error and the extended difference system, the following main result regarding reconfigured closed-loop tracking recovery can be stated and proven, which summarises the results of Lemma 11.4 and Lemma 11.5. The following theorem thus provides the solution to Problem 9.2.

**Theorem 11.1 (Stable asymptotic setpoint tracking recovery).** *Suppose that the Assumptions 9.1, 9.4, 9.3, and 9.5 are satisfied. If the LMIs (11.19) and (11.23) are feasible, then the reconfigured closed-loop system* $(\Sigma_{Pf}, \bar{\Sigma}_S, \bar{\Sigma}_A, \Sigma_C)$ *consisting of the controller (9.6), the faulty PWA system (9.7), the extended PWA virtual sensor (11.5), and the extended PWA virtual actuator (11.11) is globally ISS w.r.t. the input* $(\boldsymbol{r}, \boldsymbol{d})$*. Moreover, the output* $\boldsymbol{z}_f$ *asymptotically tracks any constant reference* $\boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$, $\bar{\boldsymbol{r}} \in \mathbb{R}^p$*, for any constant disturbance* $\boldsymbol{d}(t) = \bar{\boldsymbol{d}}\rho(t)$, $\bar{\boldsymbol{d}} \in \mathbb{R}^k$*, to nominal precision* $K \geq 0$ *in the sense that* $\limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{z}_f(t)\| \leq K$ *for all initial conditions* $\boldsymbol{x}_0$*,* $\hat{\boldsymbol{x}}_{f0}$*, and* $\boldsymbol{x}_{c0}$*.*

*The same result is achieved for arbitrary disturbances if the disturbance signal* $\boldsymbol{d}$ *is measurable, if the hypotheses of Lemma 11.1 are satisfied, and if the augmented PWA virtual sensor (11.1) and the augmented PWA virtual actuator (11.3) are used to realise the reconfiguration block* $\Sigma_R = (\bar{\Sigma}_S, \bar{\Sigma}_A)$*.*

*Proof.* See Appendix D, page 268.

Both extensions that together provide stability and tracking are based on the internal model principle. Namely, models of exo-systems creating the admissible disturbance and reference inputs have been embedded in the reconfiguration block (11.5), (11.11).

*Remark 11.2 (Occasional disturbance and reference jumps).* In practice, the extended PWA virtual sensor (11.5) still works appropriately for infrequent discontinuous changes of the disturbance input $\boldsymbol{d}$, in other words, if the disturbance is piecewise constant. Infrequent disturbance jumps may then be interpreted as changes of the initial condition of the extended observation error (11.7). "Infrequent" means that the disturbance should remain constant for several integer multiples of $1/a$, where $a$ is defined in Equation (11.20). The same considerations hold for the extended PWA virtual actuator (11.11), the reference input $\boldsymbol{r}$, and the extended difference system (11.14). ◦

*Remark 11.3 (Relation to disturbance decoupling).* The recovery of the nominal dynamical closed-loop response for arbitrary reference inputs is equivalent to a disturbance decoupling problem with stability for known disturbance (DDPS') for the reconfigured plant. To see the disturbance decoupling problem, consider the input $\boldsymbol{u}_c$ as the disturbance and the output $\boldsymbol{C}\boldsymbol{x}_{\Delta}$ (see also Fig. 3.12). The solution to this type of problem as well as to several variants of the problem are well-known for the linear case [219]. However, this is not the case for PWA systems. The solution of dynamical disturbance decoupling problems with and without stability for switching systems with known or unknown disturbances in terms of necessary and sufficient conditions remains a challenging open problem in control theory. In fact, a solution to the static DDPS' problem for PWA systems has been provided in this chapter in terms of sufficient conditions. The general and complete solution of the dynamical DDPS' problem for PWA systems remains an open problem. ◦

The design procedure for the reconfiguration block to achieve tracking is summarised in Algorithm 11.1. The steps 1-4 describe the nominal closed-loop operation before any faults occur. Once faults are detected and isolated in step 5, the design of the extended PWA virtual sensor and the extended PWA virtual actuator activates in steps 6-11, where the gains $\boldsymbol{L}$, $\boldsymbol{L}_d$, $\boldsymbol{M}$, and $\boldsymbol{M}_I$ are determined (the gain $\boldsymbol{P}$ is an additional degree of freedom that may be used to feedthrough healthy sensor measurements). After completed gain calculations, the reconfigured closed-loop system is executed in step 12, starting at reconfiguration time $t = 0$. Step 10 reduces the bump that occurs due to inaccurate guesses of the initial condition of the faulty plant.

It is straightforward to derive variations of the algorithm for the special case where the disturbances are measurable. If the relevant LMI are infeasible, then a stability-recovering and tracking-recovering virtual sensor and virtual actuator scheme might exist, but cannot be found using the sufficient conditions presented in this chapter. This problem appears to be fundamentally unavoidable, since the problem of deciding whether all trajectories of a given PWA system are bounded is undecidable [24]. In practice, it is possible to remove rows from the output matrix $\boldsymbol{C}_z$ according to a priority list until feasible solutions are found. Regarding the conservatism of common quadratic Lyapunov functions, the same considerations as in Chapter 10.3 hold.

**Algorithm 11.1.** Tracking recovering extended PWA virtual sensor and extended PWA virtual actuator synthesis

**Require:** PWA model $\boldsymbol{A}_i$, $\boldsymbol{a}_i$, $\boldsymbol{B}$, $\boldsymbol{B_d}$, $\boldsymbol{C}$, $i \in \{1,\dots,r\}$, initial time $t_0 < 0$, guessed initial condition $\hat{\boldsymbol{x}}_{f0}$

1: Initialise the nominal closed-loop system (9.1), (9.6), (11.5), (11.11) with $\boldsymbol{C}_f = \boldsymbol{C}$, $\boldsymbol{B}_f = \boldsymbol{B}$, $\boldsymbol{a}_{f,i} = \boldsymbol{a}_i$, $\boldsymbol{L} = \boldsymbol{0}$, $\boldsymbol{L}_d = \boldsymbol{0}$, $\boldsymbol{P} = \boldsymbol{I}$, $\boldsymbol{M} = \boldsymbol{0}$, $\boldsymbol{M}_I = \boldsymbol{0}$, $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$, $\boldsymbol{x}_c(t_0) = \boldsymbol{x}_{c0}$, $\hat{\boldsymbol{x}}_f(t_0) = \hat{\boldsymbol{x}}_{f0}$, $\hat{\boldsymbol{d}}(t_0) = \boldsymbol{0}$, $\tilde{\boldsymbol{x}}(t_0) = \hat{\boldsymbol{x}}_{f0}$, $\boldsymbol{x}_I(t_0) = \boldsymbol{0}$. Set the virtual actuator inactive by setting $\boldsymbol{u}_f(t) = \boldsymbol{u}_c(t)$.
2: Solve LMIs (11.19) with $\boldsymbol{C}_f = \boldsymbol{C}$ and compute a stabilising extended PWA virtual sensor gain $\bar{\boldsymbol{L}} \triangleq \bar{\boldsymbol{X}}_s^{-1}\bar{\boldsymbol{Y}}_s$, update extended PWA virtual sensor (11.5)
3: **repeat**
4: Run nominal closed-loop system
5: **until** actuator or sensor fault $f$ detected and isolated
6: Construct fault model $\boldsymbol{a}_{f,i}$, $\boldsymbol{B}_f$, $\boldsymbol{C}_f$ and update the extended PWA virtual sensor (11.5) and the extended PWA virtual actuator (11.11)
7: Solve LMIs (11.19) and (11.23) for $\bar{\boldsymbol{X}}_s$, $\bar{\boldsymbol{Y}}_s$, $\bar{\boldsymbol{X}}_a$, $\bar{\boldsymbol{Y}}_a$
8: Compute $\bar{\boldsymbol{L}} \triangleq \bar{\boldsymbol{X}}_s^{-1}\bar{\boldsymbol{Y}}_s$ and $\bar{\boldsymbol{M}} \triangleq \bar{\boldsymbol{Y}}_a\bar{\boldsymbol{X}}_a^{-1}$
9: Update the extended PWA virtual sensor (11.5) with $\bar{\boldsymbol{L}} = (\boldsymbol{L}^T\ \boldsymbol{L}_d^T)^T$
10: Wait for PWA virtual sensor to converge for specified time interval
11: Update and activate the extended PWA virtual actuator (11.11) with $\bar{\boldsymbol{M}} = (\boldsymbol{M}\ \boldsymbol{M}_I)$ and initialise $\tilde{\boldsymbol{x}}(0) = \hat{\boldsymbol{x}}_f(0)$
12: Run reconfigured closed-loop system (9.6), (9.7), (11.5), (11.11)

**Result:** Globally ISS reconfigured closed-loop system that tracks constant reference inputs in the presence of constant disturbances.

**Example 11.1 (Tracking recovering extended PWA virtual sensor and actuator synthesis for the two-tank system).** *In this example, an extended PWA virtual sensor and an extended PWA virtual actuator are designed for the two-tank system in order to reconfigure the closed-loop system after the occurrence of the sensor fault $f_s$ and the valves failure $f_{a1}$ and degradation $f_{a2}$.*

*The response of the tanks model subject to these faults is shown in Fig. 11.2. The sensor for the level $h_1$ fails at $t_{fs} = 40$ s, whereas the lower valve fails at $t_{fa1} = 20$ s and the upper valve degrades at $t_{fa2} = 35$ s. Due to the reconfiguration, it can be seen that the system follows the reference trajectory very closely. The extended PWA virtual sensor correctly estimates the state in spite of the unmeasured outflow d from tank* $\mathrm{T}_1$ *that sets in shortly after the failure of the level sensor for that tank. The extended PWA virtual actuator compensates the blocked lower valve by moving the upper valve to a compensating offset position. This example also shows that the tracking result of Theorem 11.1 is also practically useful if the reference input is not constant but piecewise constant.*

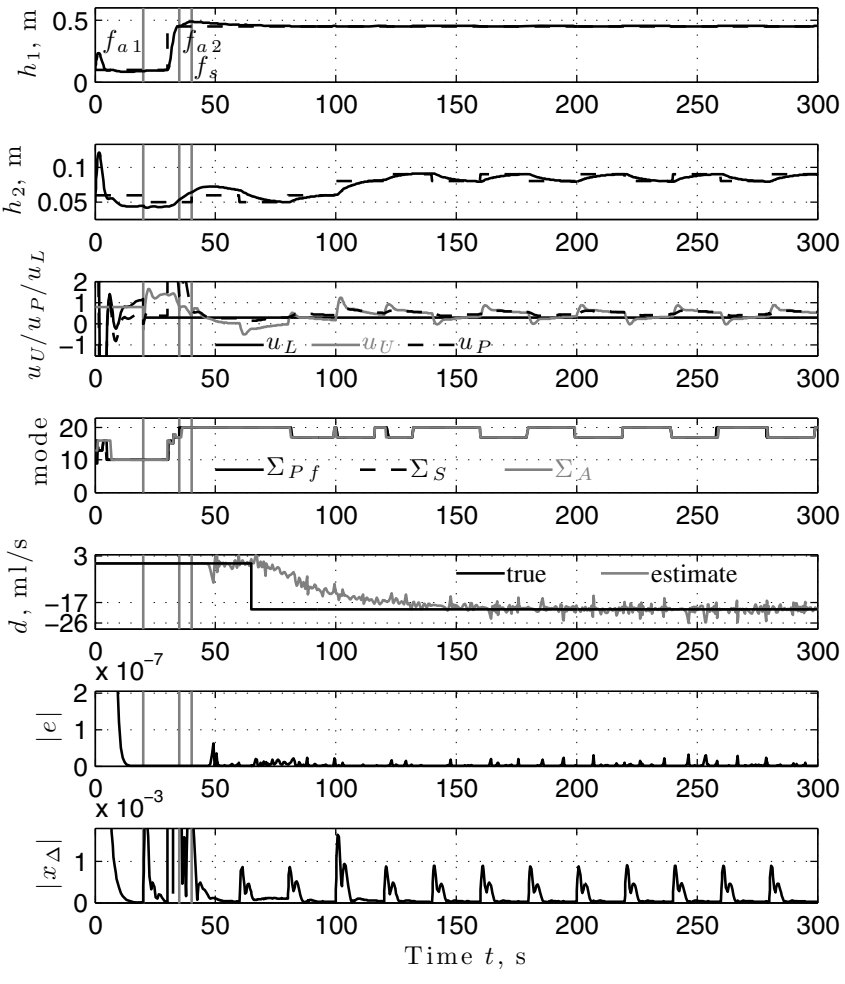


**Fig. 11.2** Response of reconfigured closed-loop tanks system based on extended PWA virtual sensor and extended PWA virtual actuator with periodic reference input.

## 11.5 Robustness against Piecewise Affine Model Uncertainties and Disturbance Variation

In the previous sections, it has been tacitly assumed that the faulty plant, the extended PWA virtual sensor, and the extended PWA virtual actuator are PWA systems, and moreover, that the disturbance is constant. This section relaxes those assumptions and studies the robustness of the reconfiguration scheme against approximation errors of the PWA model used in the reconfiguration block $\Sigma_R = (\bar{\Sigma}_S, \bar{\Sigma}_A)$. Both issues are relevant in practice because the PWA model used

in the reconfiguration block never perfectly describes the true plant dynamics in practice, and because disturbances are rarely constant.

For the following analysis, it is assumed that the faulty nonlinear system is an input-affine system of the form

$$\Sigma_{Pf,NL}: \left\{ \dot{\boldsymbol{x}}_f(t) \;= \boldsymbol{f}(\boldsymbol{x}_f(t)) + \boldsymbol{B}_f \boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}} \boldsymbol{d}(t), \right. \tag{11.26}$$

with $\boldsymbol{f}$ continuous, whereas the PWA virtual sensor (10.1) is based on the PWA model

$$\Sigma_{Pf}: \left\{ \dot{\boldsymbol{x}}_f(t) \;= \boldsymbol{A}_i \boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f \boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}} \boldsymbol{d}(t) \quad \text{for } \boldsymbol{x}_f(t) \in \Lambda_i,\ i \in \{1,\ldots,r\}. \right.$$

The difference between the input-affine model and the PWA model,

$$\boldsymbol{\varepsilon}(\boldsymbol{x}_f) = \boldsymbol{f}(\boldsymbol{x}_f) - \boldsymbol{A}_i \boldsymbol{x}_f - \boldsymbol{a}_{f,i} \text{ for } \boldsymbol{x}_f \in \Lambda_i,\ i \in \{1,\ldots,r\}, \tag{11.27}$$

represents the nonlinear approximation error. The disturbance is now assumed to be time-varying and generated by the exogenous system

$$\left\{ \dot{\boldsymbol{d}}(t) \;= \boldsymbol{\varrho}(t),\ \boldsymbol{d}(0) = \boldsymbol{d}_0, \right. \tag{11.28}$$

where the disturbance variation rate $\boldsymbol{\varrho}$ modifies the disturbance if nonzero ($\boldsymbol{\varrho} \neq \boldsymbol{0}$), and keeps the disturbance constant if zero ($\boldsymbol{\varrho} = \boldsymbol{0}$). Using the model approximation error (11.27), the input-affine system (11.26) is re-written as a perturbed PWA model

$$\Sigma_{Pf,NL}: \begin{cases} \dot{\boldsymbol{x}}_f(t) \;= \boldsymbol{A}_i \boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \boldsymbol{B}_f \boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}} \boldsymbol{d}(t) + \boldsymbol{\varepsilon}(\boldsymbol{x}_f(t)) \text{ for } \boldsymbol{x}_f(t) \in \Lambda_i, \\ \qquad\qquad i \in \{1,\ldots,r\}. \end{cases}$$

It is assumed here that the model approximation error is uniformly bounded, which is always achievable on a compact subset $\mathcal{X} \subset \mathbb{R}^n$ of the state space by sufficiently refining the state-space partition that underlies the PWA model. It is likewise assumed that the variation rate $\boldsymbol{\varrho}$ of the disturbance is globally bounded:

$$\exists E \text{ such that } \forall \boldsymbol{x}_f \in \mathcal{X} \subset \mathbb{R}^n : \|\boldsymbol{\varepsilon}(\boldsymbol{x}_f)\| \leq E \tag{11.29}$$

$$\exists F \text{ such that } \forall t \in \mathbb{R} : \|\boldsymbol{\varrho}(t)\| \leq F. \tag{11.30}$$

In order to obtain robustness for the reconfigured closed-loop system, Assumption 9.3 is replaced by the following assumption about robustness of the nominal control scheme.

**Assumption 11.1 (Robust stabilising and tracking nominal control).** *The feedback interconnection $(\Sigma_P, \Sigma_C)$ of the nominal PWA system (9.1) with bounded measurement noise $\boldsymbol{n}_{\boldsymbol{y}}$ ($\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{n}_{\boldsymbol{y}}(t)$) and the nominal controller (9.6) is ISS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{n}_{\boldsymbol{y}})$ and IOS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{d}, \boldsymbol{n}_{\boldsymbol{y}})$ and the output $(\boldsymbol{x}, \boldsymbol{u}_c)$. Furthermore, constant reference commands $\boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)$, $\bar{\boldsymbol{r}} \in \mathbb{R}^p$, are asymptotically tracked to precision $K' \geq 0$ in the presence of time-varying disturbances $\boldsymbol{d}(t)$ and*

*measurement noise* $\boldsymbol{n_y}(t)$ *(*$\boldsymbol{y}(t) = \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{n_y}(t)$ *with the property* $\lim_{t\to\infty} \boldsymbol{n_y}(t) \neq \boldsymbol{0}$*) with constant steady-state control input* $\bar{\boldsymbol{u}}_c \in \mathbb{R}^m$ *in the sense that for all* $\boldsymbol{x}_0$, $\boldsymbol{x}_{c0}$

$$\{\boldsymbol{d}(t) \text{ according to (11.28)}\,,\ \boldsymbol{r}(t) = \bar{\boldsymbol{r}}\rho(t)\} \Rightarrow \begin{cases} \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{z}(t)\| \leq K' \\ \lim_{t\to\infty} \boldsymbol{u}_c(t) = \bar{\boldsymbol{u}}_c. \end{cases}$$

Note that due to the time-varying disturbance and the persistent measurement noise, the tracking precision $K'$ is typically larger than the nominal tracking precision of Assumption 9.3. The magnitude of $K'$ will typically depend on the variation bound $E$ on $\|\boldsymbol{\varrho}\|$ as well as on a bound on measurement noise $\boldsymbol{n_y}$.

The inclusion of the model approximation error into the extended observation error (11.7) leads to the new dynamics for the extended observation error

$$\bar{\Sigma}_e : \begin{cases} \dot{\bar{\boldsymbol{e}}}(t) &= \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t) + \bar{\boldsymbol{e}}(t)) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t)) - \bar{\boldsymbol{\varepsilon}}(\boldsymbol{x}_f(t)) - \bar{\boldsymbol{\varrho}}(t) \end{cases} \tag{11.31}$$

where

$$\bar{\boldsymbol{\varepsilon}}(\boldsymbol{x}_f(t)) \triangleq \begin{pmatrix} \boldsymbol{\varepsilon}(\boldsymbol{x}_f(t)) \\ \boldsymbol{0} \end{pmatrix},\ \bar{\boldsymbol{\varrho}}(t) = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{\varrho}(t) \end{pmatrix}$$

and the extended observation error $\bar{\boldsymbol{e}}$, the extended state $\bar{\boldsymbol{x}}$, and the function $\bar{\boldsymbol{k}}_e(\cdot)$ have been defined in Equations (11.7) and (11.9). The following result is obtained.

**Theorem 11.2 (Robust extended PWA virtual sensor and extended PWA virtual actuator).** *Consider the faulty nonlinear system (11.26) reconfigured by means of the extended PWA virtual sensor (11.5) and the extended PWA virtual actuator (11.11), and suppose that Assumptions 9.1, 9.4, and 11.1 are satisfied. The reconfigured closed-loop system* $(\Sigma_{Pf,NL}, \bar{\Sigma}_S, \bar{\Sigma}_A, \Sigma_C)$ *is globally ISpS w.r.t. the input* $(\boldsymbol{r}, \boldsymbol{\varrho})$*, where* $\boldsymbol{\varrho}$ *is the disturbance variation rate. Moreover, if the reference input and the steady-state control input are constant, and if the nominal closed-loop system tracks the reference to precision* $K'$*, then the reconfigured closed-loop system tracks the reference input to degraded precision* $K' + c \cdot E + d \cdot F$*, where* $c, d > 0$*.*

*Proof.* See Appendix D, page 269.

The relevance of this result consists in the statement that the reconfigured closed-loop stability and tracking recovery properties are not lost if assumptions regarding model accuracy and constant disturbance inputs are violated. Rather, the ISpS property is always guaranteed, and the tracking accuracy degrades gradually as the model error and the disturbance variation increase. Due to the observability of the observation error from the output $\boldsymbol{y}_c$, the controller may reject the disturbance induced by the modelling error, as the following example shows.

**Example 11.2 (Tracking recovering extended PWA virtual sensor and actuator synthesis for the two-tank system).** *In this example, the extended PWA virtual sensor and an extended PWA virtual actuator are coupled with the nonlinear process model instead of a PWA process model. Therefore, the model used in the virtual sensor and virtual actuator do not exactly represent the process, requiring robustness against model uncertainties.*

*The response of the tanks model subject to the following faults is shown in Fig. 11.3. The lower valve fails at $t_{fa1} = 20$ s and the upper valve degrades at $t_{fa2} = 35$ s. In spite of the inaccurate PWA model for the nonlinear process, the system follows the reference trajectory very closely, although the transient behavior*

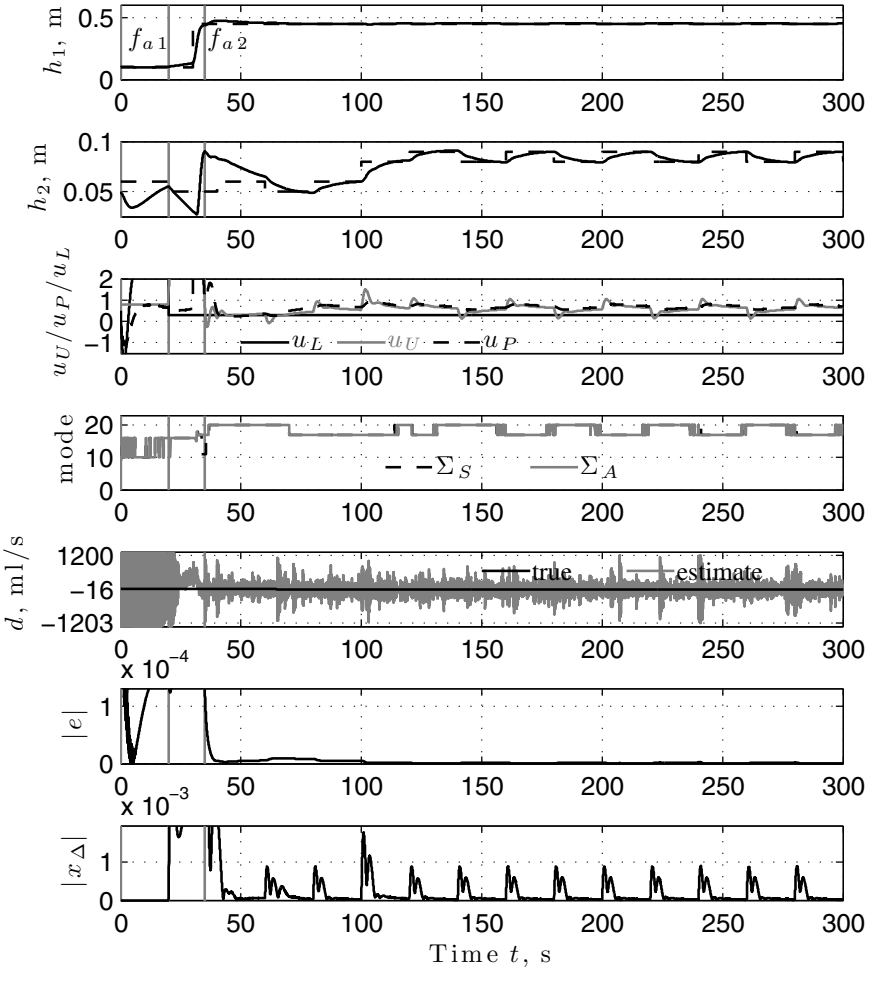


**Fig. 11.3** Response of nonlinear tanks system reconfigured using PWA model based on extended PWA virtual sensor and extended PWA virtual actuator.

*differs somewhat from the response shown in Fig. 11.2. This example also shows that the tracking result of Theorem 11.1 is robust with respect to the model approximation error. The approximation error, however, affects the disturbance estimate, which oscillates considerably between extreme values of the order of magnitude of* $\pm 1000$ *ml/s. The mean of the disturbance estimate at* $-16$ *ml/s is, however, still close to the true disturbance of* $-20$ *ml/s. The disturbance estimate provided by the extended PWA virtual sensor is, therefore, informative about the true disturbance in spite of the model uncertainties in this example.*

## 11.6 Robustness against Uncertainties of the Fault Diagnosis Result

In this section, the additional impact of uncertainties in the fault diagnosis result on the reconfiguration success is investigated, in view of the initial assumption that the fault diagnosis task has been solved. Since the model of the faulty plant is the outcome of an autonomous fault detection, isolation, and identification algorithm, this model is prone to be uncertain. In the following robustness analysis it is assumed that the model parameters $\boldsymbol{B}_f$, $\boldsymbol{a}_{f,i}$, and $\boldsymbol{C}_f$ affected by faults are subject to unknown but finite additive uncertainties $\Delta\boldsymbol{B}_f$, $\Delta\boldsymbol{a}_f$ and $\Delta\boldsymbol{C}_f$, while the model approximation error and disturbance variation introduced in the previous Section 11.5 are maintained:

$$\Sigma_{Pf,NL}: \begin{cases} \dot{\boldsymbol{x}}_f(t) &= \boldsymbol{A}_i\boldsymbol{x}_f(t) + \boldsymbol{a}_{f,i} + \Delta\boldsymbol{a}_f + (\boldsymbol{B}_f + \Delta\boldsymbol{B}_f)\boldsymbol{u}_f(t) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t) + \boldsymbol{\varepsilon}(\boldsymbol{x}_f(t)) \\ & \quad \text{for } \boldsymbol{x}_f(t) \in \Lambda_i,\ i \in \{1,\ldots,r\} \\ \boldsymbol{y}_f(t) &= (\boldsymbol{C}_f + \Delta\boldsymbol{C}_f)\boldsymbol{x}_f(t). \end{cases} \tag{11.32}$$

It is still assumed that the fault isolation task is correctly solved, so that there are no missed detections and no false positive detections.

With this uncertain plant model, the dynamics of the observation error and the difference system become

$$\bar{\Sigma}_e: \begin{cases} \dot{\bar{\boldsymbol{e}}}(t) &= \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t) + \bar{\boldsymbol{e}}(t)) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}(t)) - \bar{\boldsymbol{\varepsilon}}(\boldsymbol{x}_f(t)) - \bar{\boldsymbol{\varrho}}(t) + \boldsymbol{i}_1(t) - \boldsymbol{i}_2(t) - \Delta\bar{\boldsymbol{a}}_f \\ \boldsymbol{i}_1(t) &= \bar{\boldsymbol{L}}\Delta\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_\Delta(t) - \boldsymbol{e}(t)),\ \boldsymbol{i}_2(t) = \Delta\bar{\boldsymbol{B}}_f\bar{\boldsymbol{M}}\begin{pmatrix}\boldsymbol{x}_\Delta(t)\\ \boldsymbol{x}_I(t)\end{pmatrix} \end{cases} \tag{11.33}$$

where $\bar{\boldsymbol{e}}$ is defined in Equation (11.7), $\bar{\boldsymbol{k}}_e$ is defined in Equation (11.9), $\bar{\boldsymbol{L}}$ is defined in Equation (11.10), $\bar{\boldsymbol{M}}$ is defined in Equation (11.13), and where

$$\Delta\bar{\boldsymbol{a}}_f \triangleq \begin{pmatrix}\Delta\boldsymbol{a}_f\\ \boldsymbol{0}\end{pmatrix},\ \Delta\bar{\boldsymbol{B}}_f = \begin{pmatrix}\Delta\boldsymbol{B}_f\\ \boldsymbol{0}\end{pmatrix},$$

and

$$\bar{\Sigma}_{\Delta}: \begin{pmatrix} \dot{\boldsymbol{x}}_{\Delta}(t) \\ \dot{\boldsymbol{x}}_{I}(t) \end{pmatrix} = \bar{\boldsymbol{k}}_{\Delta}\left(\begin{pmatrix} \tilde{\boldsymbol{x}}(t) \\ \boldsymbol{x}_{I}(t) \end{pmatrix}\right) - \bar{\boldsymbol{k}}_{\Delta}\left(\begin{pmatrix} \tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) \\ \boldsymbol{0} \end{pmatrix}\right) + \begin{pmatrix} \boldsymbol{B}\boldsymbol{u}_c(t) + \boldsymbol{L}\boldsymbol{C}_f\boldsymbol{e}(t) \\ \boldsymbol{0} \end{pmatrix} + \begin{pmatrix} \boldsymbol{i}_4(t) \\ \boldsymbol{0} \end{pmatrix}, \tag{11.34}$$

where $\bar{\boldsymbol{k}}_{\Delta}$ is defined in Equation (11.12) and $\boldsymbol{i}_4(t) \triangleq \boldsymbol{L}\Delta\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t))$. Furthermore, the controller input changes to

$$\begin{aligned} \boldsymbol{y}_c(t) &= \boldsymbol{C}\tilde{\boldsymbol{x}}(t) - \boldsymbol{P}\boldsymbol{C}_f\boldsymbol{e}(t) + \boldsymbol{i}_3(t) \\ \boldsymbol{i}_3(t) &= \boldsymbol{P}\Delta\boldsymbol{C}_f(\tilde{\boldsymbol{x}}(t) - \boldsymbol{x}_{\Delta}(t) - \boldsymbol{e}(t)) \end{aligned} \tag{11.35}$$

These models show that the additive model uncertainties introduce additional feedback loops. Consequently, stability recovery requires small-gain conditions to be satisfied.

**Theorem 11.3 (Robustness against uncertain fault diagnosis).** *The reconfigured closed-loop system (9.6), (11.5), (11.11), (11.32) with the model approximation error $\boldsymbol{\varepsilon}$ and the disturbance variation ra $\boldsymbol{\varrho}$ is robustly ISpS w.r.t. the input $(\boldsymbol{r}, \boldsymbol{\varrho})$ in spite of additive diagnosis uncertainties $\Delta\boldsymbol{B}_f$, $\Delta\boldsymbol{a}_f$, and $\Delta\boldsymbol{C}_f$, if the following small-gain condition is satisfied:*

$$\left\| \begin{pmatrix} \bar{\boldsymbol{L}}\Delta\boldsymbol{C}_f & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \Delta\bar{\boldsymbol{B}}_f\boldsymbol{M} & \Delta\bar{\boldsymbol{B}}_f\boldsymbol{M}_I \\ \boldsymbol{P}\Delta\boldsymbol{C}_f & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{L}\Delta\boldsymbol{C}_f & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix} \right\| \leq \frac{1}{g}, \tag{11.36}$$

*where $g$ is the ISS gain of the feedback-interconnection $(\Sigma_C, \Sigma_{\tilde{P}}, \bar{\Sigma}_e, \bar{\Sigma}_{\Delta})$ from the input $(\boldsymbol{i}_1^T, \boldsymbol{i}_2^T, \boldsymbol{i}_3^T, \boldsymbol{i}_4^T)^T$ to the output $(\tilde{\boldsymbol{x}}^T - \boldsymbol{x}_{\Delta}^T - \boldsymbol{e}^T, \boldsymbol{x}_I^T, \boldsymbol{x}_{\Delta}^T)^T$.*

*Proof.* See Appendix D, page 270.

The small-gain conditions can be satisfied in applications in the following ways:

- Suppose that the models of non-faulty actuators and sensors are certain so that corresponding rows and columns in $\Delta\boldsymbol{B}_f$ and $\Delta\boldsymbol{C}_f$ are zero, and furthermore suppose that the LMIs (11.19) and (11.23) are feasible also if setting to zero those columns of $\boldsymbol{B}_f$ that correspond to faulty actuators, as well as setting to zero those rows of $\boldsymbol{C}_f$ that correspond to faulty sensors. In other words, faulty components are disregarded from the control loop, and consequently, corresponding rows of $\bar{\boldsymbol{M}}$ and columns of $\bar{\boldsymbol{L}}$ and $\boldsymbol{P}$ may be set to zero without changing the convergence properties. Then, the products $\Delta\bar{\boldsymbol{B}}_f\boldsymbol{M}$, $\Delta\bar{\boldsymbol{B}}_f\boldsymbol{M}_I$, $\boldsymbol{L}\Delta\boldsymbol{C}_f$, $\bar{\boldsymbol{L}}\Delta\boldsymbol{C}_f$, and $\boldsymbol{P}\Delta\boldsymbol{C}_f$ are automatically zero, and the small-gain conditions are satisfied.

- The small-gain conditions indicate that small virtual sensor gains $\bar{\boldsymbol{L}}$ and $\boldsymbol{P}$ and small virtual actuator gains $\bar{\boldsymbol{M}}$ improve robustness. Therefore, limited gains should be included as optimisation goals or additional constraints in numerical algorithms for finding solutions to the LMIs (11.19) and (11.23).

This analysis has shown that the fault-hiding principle is robust against typical fault diagnosis uncertainties that are to be expected in practice. The problem of jointly designing a fault diagnosis system and a reconfiguration strategy is a completely different topic that is beyond the scope of this monograph.

## 11.7 Duality between Extended Piecewise Affine Virtual Sensor and Virtual Actuator

The duality between the design of PWA virtual sensors and PWA virtual actuators found in the previous chapter is preserved under the extensions made in this chapter. The extended PWA virtual sensor $\bar{\Sigma}_S$ and the extended PWA virtual actuator $\bar{\Sigma}_A$ are thus dual systems in the following sense.

**Theorem 11.4 (Duality between the extended PWA virtual sensor and the extended PWA virtual actuator).** *Any solution $\bar{\boldsymbol{L}}$ to the extended PWA virtual sensor design problem for the pair $(\bar{\boldsymbol{C}}_f, \bar{\boldsymbol{A}}_i)$ also parameterises a corresponding solution $\bar{\boldsymbol{M}}$ to the PWA virtual actuator design problem for the pair $(\bar{\boldsymbol{A}}_i, \bar{\boldsymbol{B}}_f) = (\bar{\boldsymbol{A}}_i^T, \bar{\boldsymbol{C}}_f^T)$. The solutions $\bar{\boldsymbol{L}}$ and $\bar{\boldsymbol{M}}$ are linked by means of the relation $\bar{\boldsymbol{L}} = \bar{\boldsymbol{M}}^T$.*

*Proof.* See Appendix D, page 271.

The result implies that an approach for obtaining suitable gains for one system can be used to obtain suitable gains of the other system.

## 11.8 Summary and Discussion

This chapter has provided a solution to both the stability recovery problem and the setpoint tracking recovery problem for constant setpoint and constant disturbance signals after actuator and sensor faults in PWA systems, where the faults are modelled as changed input matrices, changed output matrices, and a changed affine term. The solution consists of an extended PWA virtual sensor and an extended PWA virtual actuator that also compensate blocked actuators if possible.

The main extension with respect to the PWA virtual sensor the PWA virtual actuator introduced in the previous chapter consists in the inclusion of internal models for the setpoint and disturbance signals in the reconfiguration block, which are assumed to be generated by exogenous systems with constant dynamics. The extended

PWA virtual sensor can be interpreted as an unknown-input observer for the class of PWA systems with constant disturbance. Although the theory is strictly valid only for constant input disturbance signals, it is practical also for slowly varying disturbances. The weak fault-hiding property is a consequence of the structures of the extended PWA virtual sensor and the extended PWA virtual actuator (Lemma 11.2). Further properties depend on the choice of their free parameters.

The main result is a set of sufficient conditions for reconfigured closed-loop ISS and tracking in LMI form (Theorem 11.1). The conditions are also used to formulate a procedure for determining the gains of the extended PWA virtual sensor and the extended PWA virtual actuator such that the nominal stability and tracking are recovered by the reconfigured closed-loop system (Algorithm 11.1). With respect to the requirements stated in Chapter 1.3, the method is suitable for autonomous online application. Given the current computational power, it is numerically applicable to small- and medium-size problems. The reconfiguration method is robust against inaccuracy in the model of the faulty plant, as well as robust against violations of the assumption that the disturbance signal is constant (Theorem 11.2). Furthermore, robustness against uncertain fault diagnosis results was shown (Theorem 11.3).

An open question for future research concerns the optimal choice of solutions to the LMIs (11.19) and (11.23), which each have infinite numbers of solutions. This freedom might be used to address performance requirements expressed in terms of the dynamics of the observation error system and difference system. A further open problem concerns accounting for input constraints.

A desirable relaxation concerns the continuity assumptions (Assumption 9.1 and Assumption 9.4). Covering the class of discontinuous PWA systems would increase the scope of possible applications for the reconfiguration approach. However, the convergence property has been used in this chapter to guarantee tracking, and the existence of common quadratic Lyapunov functions is known to be insufficient for the convergence of discontinuous PWA systems with more than two modes [157]. The extension to discontinuous systems for stronger goals than stability is therefore a challenging open problem.

This chapter concludes Part III on the reconfigurable control of PWA systems, whose consideration is motivated by their ability to approximate nonlinear dynamical systems to far better accuracy than linear systems. The stability and tracking recovery problems have been solved, and certain robustness and duality properties hold.

The end of Part III also closes the parts of theoretical contributions to reconfigurable control theory presented in this monograph. As regards the expressiveness of the different modelling frameworks used in this monograph, Tables 4.1 to 9.1 have shown in retrospect that no single class of system models captures every relevant technological fault aspect. This fact was highlighted through the ship example. This observation emphasises the need for having developed reconfigurable control approaches for each system class separately. The following final Part IV discusses practical applications of the theoretical results developed in Chapters 5 to 11 in experimental environments.

# Part IV
# Applications

In this part, applications of the nonlinear fault-hiding approaches to reconfigurable control that were developed in the previous parts are described. First, it is explained how the fault-hiding approach with its reconfiguration blocks should be included into an embedded real-time control framework, and a rapid-prototyping implementation is described. Next, the application of the fault-hiding approach to a benchmark thermofluid process is described. The process has been implemented on the pilot plant VERA at the Institute of Automation and Computer Control in Bochum. Applications of the aforementioned methods are summarised for several fault cases.

# Chapter 12
# Application Framework

**Abstract.** This chapter explains how the fault-hiding approach to reconfigurable control is implemented in a real-time control framework. The required information flow between plant, controller, and reconfiguration block is described, and it is shown how reconfiguration blocks can be embedded into modern control hardware. A MATLAB toolbox providing prototype implementations of reconfiguration blocks is described. Finally, possible applications for virtual actuators and virtual sensors outside the field of fault-tolerant control are sketched.

## 12.1 Information Structure of a Real-Time Control Framework

This section describes the required information flow between sensors, the controller, actuators, and the reconfiguration block in order to enable reconfigurable control according to the methods presented in this monograph. For the matter of this section, the implementation details regarding control hardware, communication protocols, and communication architectures are irrelevant.

Figure 12.1 shows the required structure of a reconfigurable closed-loop system. The plant is equipped with actuators and sensors. The combination of physical plant, actuators, and sensors was called "plant" $\Sigma_P$ (nominal) or $\Sigma_{Pf}$ (faulty) in Parts I–III of this monograph (grey box in Fig. 12.1). It is important that the models also represent those actuators and sensors that are not used by the nominal controller. In other words, the plant models must be complete with respect to the installed redundancies, which can otherwise not be discovered by the reconfiguration algorithms. In Fig. 12.1, the block labelled "physical plant" refers to the physical system. The controller is arbitrary, in particular, it may be centralised or decentralised, linear or nonlinear, as long as the nominal closed-loop system satisfies the assumptions of the previous chapters.

The key aspect of the implementation is the immediate inclusion of the reconfiguration block, which is a permanent part of the closed-loop system, whether or not faults have occurred. Its permanence is enabled by the inactivity conditions that every reconfiguration block must satisfy according to Definition 3.5. The inactivity

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 193–202.
springerlink.com 

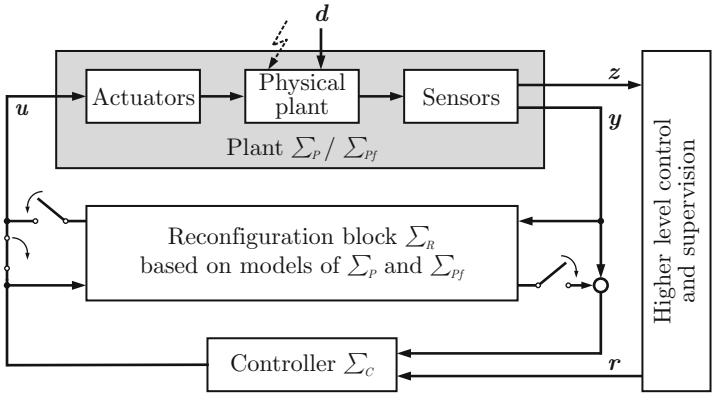


**Fig. 12.1** Information flow in a real-time control framework with reconfiguration block.

conditions ensure that, before the occurrence of faults, the closed-loop system with reconfiguration block behaves exactly like the nominal closed-loop system without the reconfiguration block. The inactivity is symbolised in Fig. 12.1 by means of switches. Before the occurrence of a fault, the switches are in the shown positions, and the reconfiguration block has no influence on the closed-loop system. Note that the signals entering the reconfiguration block directly without passing through switches are important so that an observer inside the reconfiguration block can track the state of the plant also while the reconfiguration block is inactive. The inactivity conditions imply that the switches shown in Fig. 12.1 do not need to be implemented.

Once a fault has been detected and isolated, the parameters of the reconfiguration block are updated, and the reconfiguration block is activated by moving the switches in the directions indicated in Fig. 12.1. The synthesis of its parameters is based on models $\Sigma_P$ and $\Sigma_{Pf}$ of the physical plant including its actuators and sensors. In the context of this monograph, the model can be a linear model, a Hammerstein-Wiener model, or a piecewise affine model. The model types were introduced and discussed in Chapters 4, 5, and 9. Synthesis methods for linear reconfiguration blocks have been summarised in Chapter 4. The possible synthesis methods for the parameters of Hammerstein-Wiener reconfiguration blocks have been provided in Chapters 6 to 8. Synthesis methods for the parameters of piecewise affine reconfiguration blocks have been provided in Chapters 10 and 11.

Figure 12.1 furthermore shows that reference commands are typically generated by a high-level control component with supervisory tasks. The high-level component may be human or automatic. However, the presence of the higher level gives the fault hiding approach a further dimension. In the previous parts of this monograph, the term ”fault-hiding” refers to the comparison of the nominal and reconfigured plant behaviour as seen from the controller. The fault-hiding idea can also be applied to the interface between the supervisory level and the control level. Namely, a successful reconfiguration at the control level that well recovers the closed-loop

behaviour from reference $\boldsymbol{r}$ to output $z$ will hide the presence of faults and ease the task of the supervisory level in responding of the fault. Fault-hiding at the supervisory level is exactly achieved if and only if performance recovery is exactly achieved at the control level.

## 12.2 Physical Realisation within Modern Control Architectures

The reconfiguration block must be implemented with real-time constraints similar or equal to the real-time properties of the nominal controller on embedded real-time control hardware. Embedded real-time control hardware refers to programmable logic controllers (PLC), electronic control units (ECU), and similar devices. The implementability is mainly affected by the availability of three elements:

- implementations of computational operations,
- computational power,
- storage capacity.

These elements are now discussed based on typical computation and storage capacities of contemporary embedded real-time control hardware[1]. Since such hardware is subject to quick development towards more storage capacity and more computational power, the following statements are only of contemporary validity.

The implementation of the Hammerstein-Wiener virtual actuator and virtual sensor requires the ability to perform numerical integration, addition, multiplication, and the evaluation of nonlinear input and output functions, which might be implemented by means of table-lookup and interpolation. The implementation of the PWA virtual actuator and sensor requires the same abilities, where the nonlinear functions are PWA. The described computational operations are available in contemporary embedded real-time control hardware. The implementation of linear, Hammerstein-Wiener, and PWA reconfiguration blocks is therefore practically feasible from a functional perspective.

Regarding the computational power, the implementability depends on the speed of the processor, on the dynamical model order, and on the required sampling time. The latter depends on the plant dynamics. A general statement is impossible, but as a general rule, the methods are implementable for moderately fast plant dynamics on standard computing hardware. Since all methods presented in this monograph are formulated in continuous time, a sufficiently high oversampling rate (such as $20\times$ the fastest plant mode) is necessary so that the control laws can be implemented in discrete time without respecting discrete-time theory [117]. While the time-discretisation of linear reconfiguration blocks is straightforward in principle, the translation of a given continuous-time nonlinear dynamical system into an equivalent discrete-time system is not straightforward due to the lack of a universally suitable and accepted method. Nevertheless, some approaches to the approximation of

[1] Electronic control units have 32Bit CPUs with up to 2MB flash memory and up to 80 MHz core clock frequency at the time of writing [62].

continuous-time control by means of discrete-time control for nonlinear systems exist, see [71]. For this reason, the quasi-continuous implementation of the nonlinear reconfiguration block by means of oversampling is recommended.

The memory requirements for storing Hammerstein-Wiener models are slightly larger than for linear models due to the presence of nonlinear functions. The burden related to evaluating these functions can be partly moved from the processor to memory by replacing online function evaluation with table lookup. The memory requirements for storing PWA models depend heavily on the desired model accuracy. If high numbers of polytopes are required to achieve satisfactory model accuracy, then the memory burden may become prohibitive.

In summary, the online implementation of reconfiguration blocks with fixed parameters is practically feasible on contemporary embedded real-time control hardware for medium-size systems with moderately fast dynamics.

However, the synthesis of the reconfiguration block parameter imposes a considerable computational burden that is too high for contemporary embedded real-time controller, since it requires the solution of large systems of LMI. Furthermore, their solution requires complex numerical methods that are generally unavailable on such hardware. It is, however, not necessary to collocate the synthesis of reconfiguration block parameters with its real-time execution on the same hardware. It is generally recommended to separate the supervision tasks such as fault diagnosis and reconfiguration block synthesis from real-time control execution, as shown in Fig. 12.2.

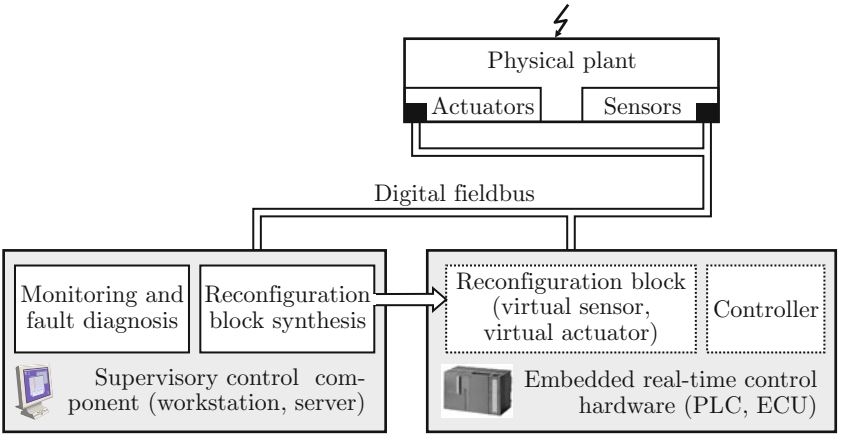


**Fig. 12.2** Hardware architecture of a real-time control framework with reconfiguration block.

Figure 12.2 shows a physical plant with actuators and sensors, embedded real-time control hardware, and a supervisory control component, which are linked over a digital fieldbus. The nominal controller and reconfiguration block are implemented and executed on the embedded real-time control hardware. This architecture requires deciding on the class of the used reconfiguration block before process startup. The inactivity conditions ensure that the closed-loop behaviour is not affected by

the reconfiguration block prior to the occurrence of faults. The supervisory control component (implemented on a workstation or server, for example) monitors the measured and manipulated signals and performs fault diagnosis tasks (FDI). If a fault is detected and isolated, the supervisory control component computes suitable parameters for the reconfiguration block and communicates these parameters to the embedded control hardware as indicated by the broad arrow, which updates the reconfiguration block parameters, and thereby activates the reconfiguration block. By following through these steps, control reconfiguration is achieved.

Furthermore, it is not necessary to collocate the nominal controller and the reconfiguration block on the same embedded real-time control hardware. Two (or more, in the case of decentralised nominal control) embedded controllers may be used, as long as the information flow indicated in Fig. 12.1 is achieved. Digital fieldbus networks (such as Profibus, Profinet, FoundationFieldbus, CAN, and FlexRay) [166] are nowadays used for numerous reasons, but they are also advantageous for adding reconfigurable control by means of reconfiguration blocks to existing controllers without any physical rewiring. In digital fieldbus networks, changing the signal flow does not require physical rewiring. In particular, it is possible to achieve fault-tolerance by adding new real-time hardware that implements the reconfiguration block without changing the existing control laws. This aspect is important when adding fault-tolerance to existing plants and control schemes with proven functionality.

When seeking to improve plant dependability by means of reconfiguration blocks, it should be kept in mind that every additional system component increases the overall system complexity and might become a single point of failure. Redundant computational hardware and redundant communication channels may be necessary to achieve overall system dependability.

## 12.3 Rapid Prototyping Toolbox for MATLAB and Simulink

The virtual actuators and virtual sensors described in this monograph have been implemented as Simulink blocks in a MATLAB toolbox for reconfigurable control called *CoRe* whose Simulink library is shown in Fig. 12.3. The virtual sensor is available for linear systems (Definition 4.4), for systems with saturations in the input and output paths (a special case of Definition 6.1 where $\boldsymbol{\varphi} = \text{sat}$ and $\boldsymbol{h} = \text{sat}$), and for piecewise affine systems (Definition 10.1 and Definition 11.3). The virtual actuator is available for linear systems (Definition 4.5), for systems with input saturations (Definition 7.1 where $\boldsymbol{\varphi} = \text{sat}$), and for piecewise affine systems (Definition 10.2 and Definition 11.4). Every block has a mask where the model parameters are entered, and where the desired design method is chosen. The available toolbox functionality is briefly described below, details are found in the toolbox documentation [168].

The **linear virtual actuator** block offers the following design methods:

- Pseudoinverse method (see [33])
- Stability recovery (Theorem 4.9)
- Setpoint tracking recovery, zero placement (Theorem 4.10)

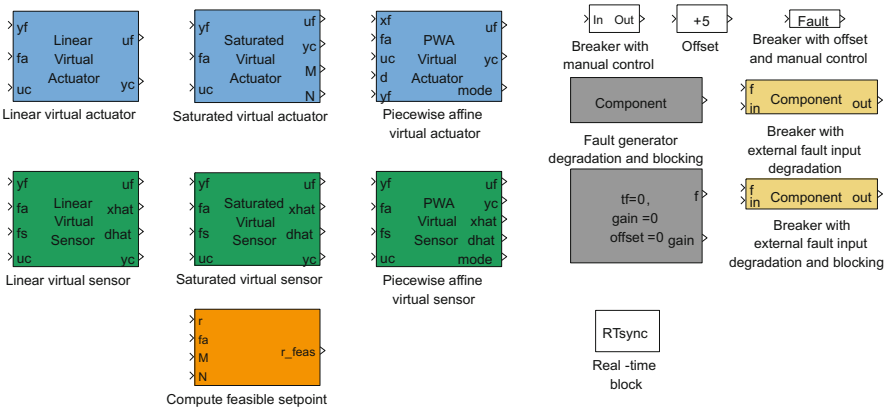


**Fig. 12.3** *CoRe*: a MATLAB toolbox for rapid prototyping of reconfiguration blocks.

- Setpoint tracking recovery, integrators (see [202])
- Exact performance recovery, Markov parameters (see [171, 175])
- Exact performance recovery (Theorem 4.11)
- Approximate performance recovery (see [180])
- Optimal performance recovery (Theorem 4.14)
- Approximate performance recovery, minimum output correction (see [175])
- Automatic choice of the strongest solvable reconfiguration problem.

It has been shown that the pseudoinverse method is not reliable. It cannot guarantee stability [202] and may even fail to close the loop after an actuator failure although trivial reconfiguration solutions exist [175].

The **linear virtual sensor** block provides the following design methods:

- Pseudoinverse method (see [33])
- Stability recovery (Theorem 4.2)
- Setpoint tracking recovery, zero placement (Theorem 4.3)
- Setpoint tracking recovery, integrators (see [202])
- Exact performance recovery, Markov parameters (see [171, 175])
- Exact performance recovery (Theorem 4.4)
- Approximate performance recovery (see [180])
- Optimal performance recovery (Theorem 4.7)
- Automatic choice of the strongest solvable reconfiguration problem.

The **saturated virtual actuator** block offers the following design methods:

- Stability and setpoint tracking recovery, Kalman-Yakubovich equations (see [172], Theorem 7.1)
- Stability and setpoint tracking recovery, LMI (Corollary 6.1, Theorem 7.1)
- Stability and optimal performance recovery (Theorem 8.3).

The **saturated virtual sensor** block provides the following design methods:

- Stability and setpoint tracking recovery, Kalman-Yakubovich equations (see [172], dual to Theorem 7.1)
- Stability recovery, LMI (Corollary 6.2)
- Stability and optimal performance recovery (dual to Theorem 8.3).

The saturated virtual actuator and saturated virtual sensor are special cases of their general Hammerstein-Wiener counterparts. For given specific nonlinear functions, the provided blocks can be easily modified by editing the underlying implementation code.

The **PWA virtual actuator** block implements the following design methods:

- Stability recovery (Theorem 10.1)
- Stability and setpoint tracking recovery, mode-dependent feedforward gains
- Stability and setpoint tracking recovery, common feedforward gains
- Stability and setpoint tracking recovery (Theorem 11.1).

The **PWA virtual sensor** block offers the following design methods:

- Stability recovery (Theorem 10.1)
- Stability and setpoint tracking recovery, mode-dependent feedforward gains
- Stability and setpoint tracking recovery, common feedforward gains
- Stability and setpoint tracking recovery (Theorem 11.1).

Furthermore, the toolbox contains blocks for computing feasible setpoints from infeasible setpoints (Algorithm 7.1) as well as fault generators and signal breakers that simplify the intentional introduction of faults into simulations and experiments. The representable faults include degradation by means of gain reduction, offsets, and complete failure with or without offsets, according to Equations (4.12), (4.13) as well as Equation (9.8). The fault generators and signal breakers are applicable to actuator and sensor signals alike.

Figure 12.4 shows a typical block diagram for fault-tolerant control with the fault-hiding approach. The reconfiguration block consists of a PWA virtual actuator and a PWA virtual sensor in this example, which have to be connected as shown. The fault is provoked by means of actuator fault generators and sensor fault generators, which trigger signal breakers. In reality, the faults are, of course, not artificially generated, but occur autonomously and spontaneously. The information about the actuator and sensor faults is provided by a fault diagnosis block as indicated in Fig. 12.4.

## 12.4 Further Applications

The previous sections of this chapter have explained how to use reconfiguration blocks in an online control reconfiguration framework. However, the fault-hiding concept that lead to virtual actuators and virtual sensors is also applicable in other fault-tolerant control frameworks as well as in application domains completely outside fault-tolerant control.

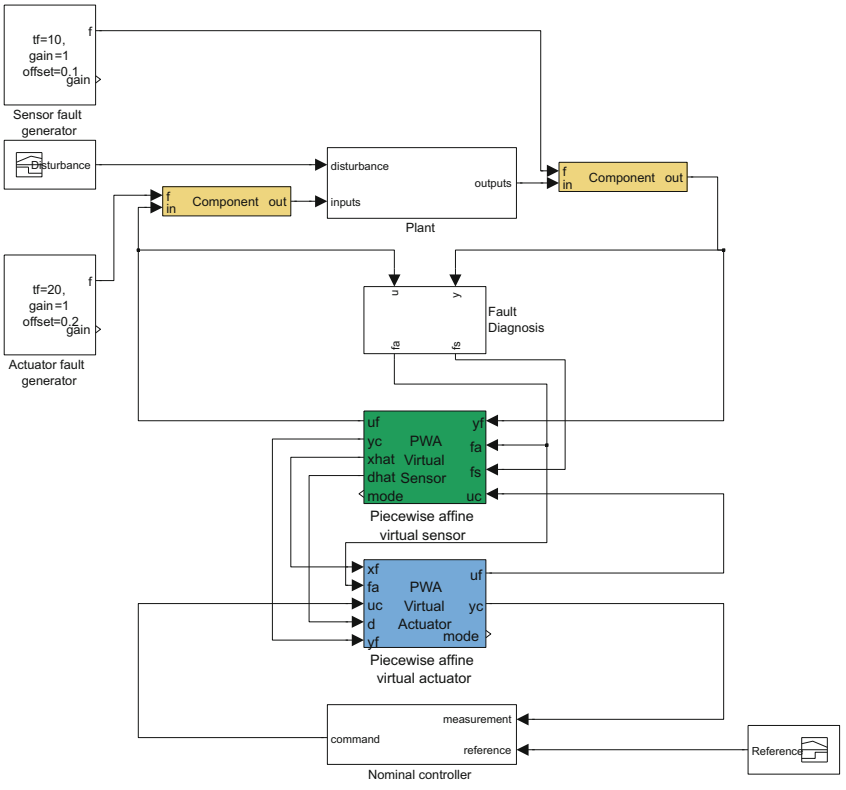


**Fig. 12.4** Typical block diagram for fault-tolerant control with the fault-hiding approach.

First, reconfiguration blocks can also be used in fault-tolerant control schemes where reconfigured controllers are designed offline and where the online part of reconfiguration reduces to the selection of the appropriate reconfigured controller from a database, as shown in Fig. 12.5. These so-called projection approaches are used if exhaustive testing of all controllers by means of extensive simulations is necessary. Exhaustive controller tests are customary in aviation and nuclear power industries, for example. The fault-hiding framework provides clear problem formulations, reconfigurability analysis techniques with necessary and sufficient conditions for the linear case as well as sufficient conditions for the nonlinear cases, and a structure of the reconfigured controller that has an intuitive interpretation as the difference between nominal and reconfigured behaviours. Therefore, it may be useful to apply the fault-hiding approach in offline synthesis problems as well as in online synthesis problems.

Second, the concept of virtual actuators is also useful in control problems that are completely outside of the domain of fault-tolerant control. Controller synthesis is relatively easy to achieve with full access to the state derivatives through suitable actuators. On the other hand, many physical plants are classified as so-called

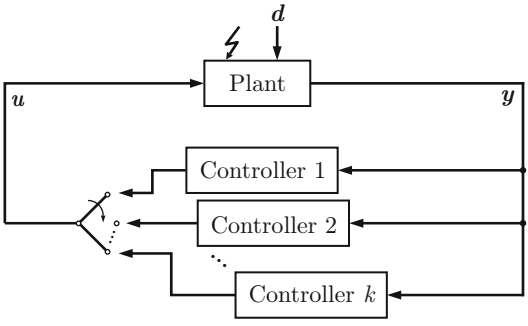


**Fig. 12.5** Selection of pre-designed controllers for every fault case.

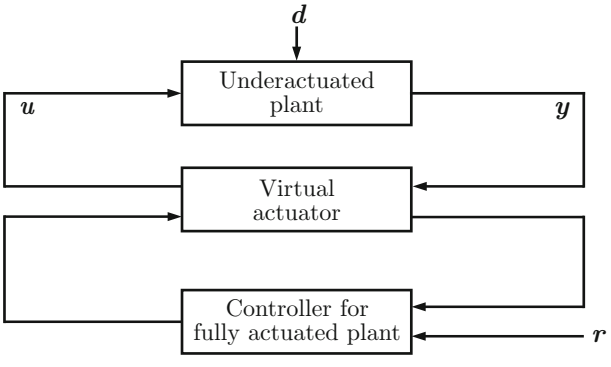


**Fig. 12.6** Control of underactuated systems by means of virtual actuators.

underactuated systems, which simply means that the control vector does not span the entire state space. The ship used as a running example in this monograph is an example of an underactuated system. The thruster and rudder actuators permit the direct control of surge and yaw velocity, but the sway velocity is not directly actuated (unless bow and stern thrusters are installed, which are usually used only in harbour maneuvers). Virtual actuators can simplify the controller synthesis as follows. First, a controller is designed for the fictitious fully actuated system with hypothetical further actuators (such as bow thrusters in the ship example). To implement this control law on the underactuated system, a virtual actuator is used, which is based on two models. The first model corresponds to the nominal system (in the terminology of fault-tolerant control) or to the fully actuated system (in the terminology of underactuated systems). The second model corresponds to the faulty system (in the terminology of fault-tolerant control) or to the underactuated system (in the terminology of underactuated systems). The virtual actuator translates between these two different systems. From this point of view, actuator faults change a given system from richer actuation to sparser actuation.

The problem of controlling distributed parameter systems described by partial differential equations (PDEs) is closely related to the control of underactuated systems. Usually, the physical plant is actuated in a finite number of points, which is called *point-wise control* or *boundary control* in the literature on distributed parameter control (see, for example, [107]). A spatial discretisation of the PDE typically leads to a large system of first-order differential equations representable in state-space form. Classical controller synthesis techniques are in principle applicable to the state-space system. With point-wise control, typically an underactuated system results, and the virtual actuator can be used to implement the controller designed for the fully actuated system. This approach has been successfully applied [36, 167] to a pipeline system described by the semi-linear transport equation [73].

# Chapter 13
# Fault-Tolerant Control of a Thermofluid Process

**Abstract.** This chapter shows simulated and experimental applications of the reconfigurable control methods presented in this monograph to a thermofluid process that has been realised on the test bed plant VERA. Closed-loop trajectories of the reconfigured process are discussed for various fault scenarios. They demonstrate that the approach is indeed suitable for solving reconfigurable control problems in practice.

## 13.1 Pilot Plant VERA

To test the control reconfiguration methods described in this monograph, the chemical pilot plant VERA (German for **Ver**fahrenstechnische Pilot**a**nlage) at the Institute of Automation and Computer Control is used (Fig. 13.1). The plant is used for the experimental evaluation of automation and control methods in a realistic, industrial environment. In this monograph, a thermofluid process implemented on VERA is used as a benchmark process.

Its process hardware consists of 8 tanks with a complex pipe system as well as 64 sensors and 82 actuators. Standard industrial components are used throughout the plant. The tanks differ with respect to their usability in processes. The educt tanks T1-T4 in the upper part provide raw substances or store intermediate products. They are connected to reactors TB, TM and TS located below via pipes. The reactors TB and TS are equipped and instrumented for performing exothermal as well as endothermal reactions. Reactor components comprise stirrers, heaters, coolers and various sensors. The tank TM is exclusively assigned to blending or storage tasks. Waste water is stored in the tank TW.

To control the medium flow, discrete cut-off valves, continuous control valves (all pneumatically driven), electrical centrifugal pumps and a pneumatically driven membrane pump are available. By means of numerous sensors, physical quantities such as level, flow, temperature, electrical conductivity and pH-value are measurable at locations of interest.

Plant automation, control and supervision are based on two industrial programmable logic controllers (PLCs) of the type SIMATIC S7-300, which host a plant

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 203–228.
springerlink.com 

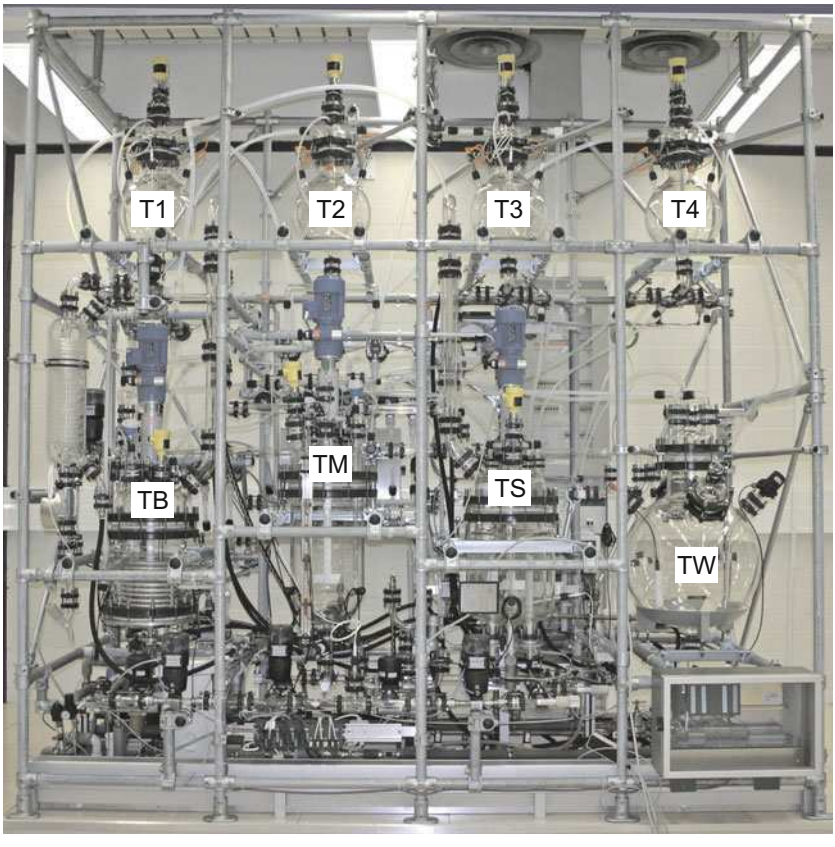


**Fig. 13.1** Chemical pilot plant VERA at the Institute for Automation and Computer Control in Bochum.

protection system [85] as well as subordinate control loops, and provide an interface to rapid control prototyping software. Rapid control prototyping is based on MATLAB and Simulink, while process monitoring and manual operation are provided by Simatic WinCC software, each implemented on dedicated workstations. The connection between MATLAB and the PLCs is a custom solution described in [59, 60] that is based on the internet standard protocol UDP over IP.

In view of Chapter 12, the nominal controller, the reconfiguration block, and the process supervision tasks are all collocated on the rapid prototyping workstation running MATLAB and Simulink.

## 13.2 Thermofluid Process

### *Process Description*

The thermofluid process illustrated in Fig. 13.2 is implemented on the pilot plant VERA. This process is used for evaluating the practical usefulness of the control reconfiguration methods developed in this monograph. The process purpose necessitates the regulation of

- fluid level $l_{\mathrm{TS}}$,
- temperature $\vartheta_{\mathrm{TS}}$, and
- electrical conductivity $\nu_{\mathrm{TS}}$

of a salt water solution in the reactor TS. In the nominal situation, three decentralised main control loops solve this task, which use the actuators best suited for solving the respective control task.

The electrical conductivity $\nu_{\mathrm{TS}}$ is regulated to its setpoint $r_{\nu,\mathrm{TS}} = 2.1$ mS/cm by means of the control valve $u_{\mathrm{TB}}$, which manipulates the salt water mass flow rate $\dot{m}_{\mathrm{TB}}$ from reactor TB to reactor TS and has an actuation range $u_{\mathrm{TB}} \in [0, 1]$. The electrical conductivity of the liquid in TB is $\nu_{\mathrm{TB}} = 4$ mS/cm. The desired electrical conductivity in reactor TS results from blending the salt concentrate from TB with cold water at the mass flow rate $\dot{m}_{\mathrm{CW}}$, which is manipulated by means of the control valve input $u_{\mathrm{CW}}$ via a subordinate control loop. The setpoint is nominally fixed to 0.04 kg/s, but potentially variable, so that the controlled cold water supply is viewed as an actuator $u_{\mathrm{CW}}$. The level $l_{\mathrm{TS}}$ is kept at its setpoint $r_{l,\mathrm{TS}} = 0.335$ m using the variable speed $u_{\mathrm{PS}}$ of the pump PS with the actuation range $u_{\mathrm{PS}} \in [0, 1]$, which empties the reactor TS into the waste-water tank TW. The temperature $\vartheta_{\mathrm{TS}}$ is regulated to the setpoint $r_{\vartheta,\mathrm{TS}} = 25\,^{\circ}\mathrm{C}$ using the electrical heater input $u_{\mathrm{el,TS}}$ and the actuation range $u_{\mathrm{el,TS}} \in [0, 1]$. The control valve $u_{\mathrm{TM}}$ for controlling the mass flow rate $\dot{m}_{\mathrm{TM}}$ remains closed in the nominal process. It provides access to the redundant source of salt concentrate in the reactor TM. Redundancies are described in more detail below.

Two further subordinate control loops keep the levels in the salt concentrate tanks TB and TM constant at $l_{\mathrm{TB}} = 0.41$ m and $l_{\mathrm{TM}} = 0.31$ m by means of top up from educt tanks. The temperature in TB is furthermore regulated to the fixed setpoint $\vartheta_{\mathrm{TB}} = 25\,^{\circ}\mathrm{C}$ using the heater input $u_{\mathrm{el,TB}}$ in the reactor TB in order to compensate for the cooling in the unheated educt tanks. These subordinate loops are only auxiliary to the process of interest. They enable longer process operation, which is limited due to the finite tank capacities.

This process has been used to evaluate the linear virtual actuator [177, 178], and it has been used in several student projects [1, 101, 102, 131, 149, 162, 188, 190, 218]. In the prior simulations and experiments, it was found that linear virtual actuators are capable of successful reconfigurable control only for a very limited number of the possible actuator fault scenarios. Models of the thermofluid process and nominal controllers are discussed in the following sections. Afterwards, the nominal control scheme and the fault scenarios are defined.

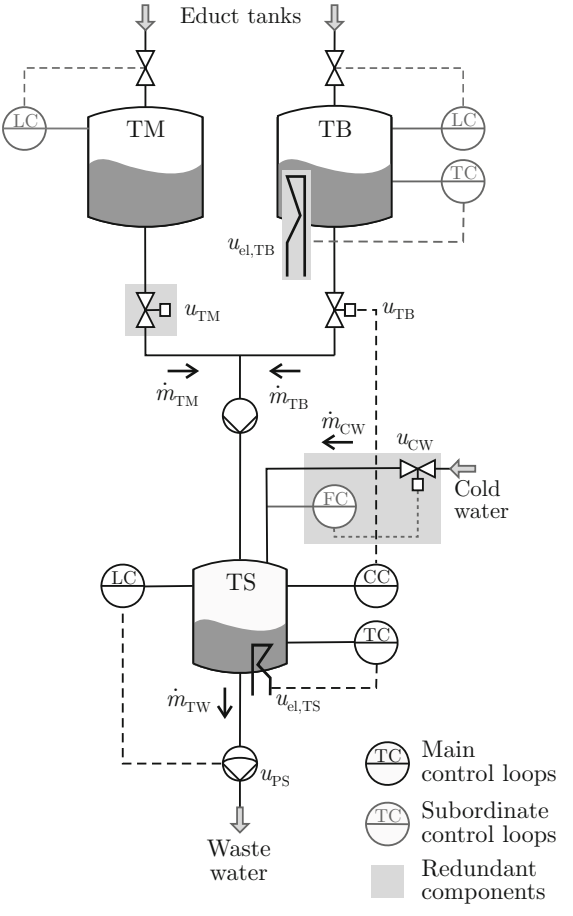


**Fig. 13.2** Flow diagram of the thermofluid process with redundant components.

## *Process Models*

In this section the process models are described. The structure of the nonlinear state-space model is given, and the complete linearised state-space model is provided. The complete nonlinear model with physical parameters is given in Appendix E.1. The nonlinear model and its linear approximation have been derived in [188] and first published in [177, 178]. The piecewise affine model was derived from the nonlinear model in [1].

**Nonlinear state-space model.** The nonlinear state-space model of the form (3.2) is based on the following definitions of the state $\boldsymbol{x}(t) \in \mathbb{R}^7$, the input $\boldsymbol{u}_c(t) \in \mathbb{R}^6$, the disturbance $\boldsymbol{d}(t) \in \mathbb{R}$, and the output $\boldsymbol{y}(t) \in \mathbb{R}^4$:

$$\boldsymbol{x}(t) = \begin{pmatrix} \vartheta_{\mathrm{TS}}(t) \\ l_{\mathrm{TS}}(t) \\ c_{\mathrm{TS}}(t) \\ \vartheta_{\mathrm{TB}}(t) \\ x_{\mathrm{CW}}(t) \\ x_{\mathrm{el,TB}}(t) \\ x_{\mathrm{el,TS}}(t) \end{pmatrix}, \quad \boldsymbol{u}_c(t) = \begin{pmatrix} u_{\mathrm{TM}}(t) \\ u_{\mathrm{TB}}(t) \\ u_{\mathrm{el,TB}}(t) \\ u_{\mathrm{el,TS}}(t) \\ u_{\mathrm{CW}}(t) \\ u_{\mathrm{PS}}(t) \end{pmatrix}, \quad \boldsymbol{d}(t) = \begin{pmatrix} 0 \\ 0 \\ d_{\lambda}(t) \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad \boldsymbol{y}(t) = \begin{pmatrix} \vartheta_{\mathrm{TS}}(t) \\ l_{\mathrm{TS}}(t) \\ \nu_{\mathrm{TS}}(t) \\ \vartheta_{\mathrm{TB}}(t) \end{pmatrix}.$$

Instead of the electrical conductivity $\nu_{\mathrm{TS}}$, the added salt concentration $c_{\mathrm{TS}}$ is used as a state variable, which is connected with the electrical conductivity via a static relationship. Further state variables are $x_{\mathrm{CW}}$, $x_{\mathrm{el,TB}}$ und $x_{\mathrm{el,TS}}$ representing the dynamics of actuation systems with first-order delay character. The disturbance represents unmodelled and unknown variations of the salt concentration in the supply container TB. The model has the form (3.2), which is the basis for the derivation of a linearised perturbation model around the nominal setpoint, as well as a piecewise affine model.

**Linear model.** A linearisation of the nonlinear state-space model (3.2) around the equilibrium given in Table E.2 of Appendix E.1 leads to a linear state-space model with the matrices

$$\boldsymbol{A} = 10^{-3} \begin{pmatrix} -3.46 & 0 & 0 & 1.47 & -59.12 & 0 & 39.36 \\ 0 & -0.76 & 0 & 0 & 1.41 & 0 & 0 \\ 0 & 0 & -3.16 & 0 & -0.0034 & 0 & 0 \\ 0 & 0 & 0 & -1.34 & 0 & 157.46 & 0 \\ 0 & 0 & 0 & 0 & -270.27 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -37.04 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -15.38 \end{pmatrix}, \tag{13.1}$$

$$\boldsymbol{B} = 10^{-3} \begin{pmatrix} -10.62 & 0 & 0 & 0 & 0 & 0 \\ 7.11 & 8.5 & 0 & 0 & 0 & -1.98 \\ 0.0249 & 0.0235 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 270.27 & 0 \\ 0 & 0 & 37.04 & 0 & 0 & 0 \\ 0 & 0 & 0 & 15.38 & 0 & 0 \end{pmatrix}, \quad \boldsymbol{B}_{\boldsymbol{d}} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

$$\boldsymbol{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2047.7 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}, \quad \boldsymbol{C}_z = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2047.7 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

The system matrix has the stable eigenvalues shown on the diagonal of matrix $\boldsymbol{A}$ given in Equation (13.1).

The linear state-space model is a much simplified process representation that naturally neglects any nonlinearities. The most relevant nonlinearities are actuator saturations, followed by quantisation of sensor measurements and actuator inputs, and dead time. The quantisation scheme of control valve inputs is furthermore

nondeterministic in reality. Near the equilibrium, the linear model is valid up to saturations, minor quantisation effects, dead zones and delays.

**Hammerstein-Wiener model.** The Hammerstein-Wiener model (5.1) for the thermofluid process has the same linear dynamics as the linear model derived in the previous section, augmented by the actuator constraints that are modelled as saturations of the input signals at lower bounds $\underline{\boldsymbol{u}}$ and upper bounds $\overline{\boldsymbol{u}}$, defined relative to the linearisation equilibrium point in Table 13.1.

**Table 13.1** Thermofluid process actuator constraints expressed as lower and upper saturation bounds relative to the linearisation equilibrium.

| Actuator | Lower bound $\underline{u}$ | Upper bound $\overline{u}$ |
|---|---|---|
| $u_{\mathrm{TM}}$ | 0 | 1 |
| $u_{\mathrm{TB}}$ | −0.16 | 0.84 |
| $u_{\mathrm{el,TB}}$ | −0.0007 | 0.9993 |
| $u_{\mathrm{el,TS}}$ | −0.6 | 0.4 |
| $u_{\mathrm{CW}}$ | −0.04 | 0.06 |
| $u_{\mathrm{PS}}$ | −0.53 | 0.47 |

**Piecewise affine model.** A PWA model of the form (9.1) has been derived from the nonlinear model of the form (3.2). More precisely, the nonlinear model is of the input-affine form

$$\begin{cases} \dot{\boldsymbol{x}}(t) &= \boldsymbol{f}(\boldsymbol{x}(t)) + \boldsymbol{g}(\boldsymbol{x}(t))\boldsymbol{u}(t) + \boldsymbol{B}_d\boldsymbol{d}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) \\ \boldsymbol{z}(t) &= \boldsymbol{C}_z\boldsymbol{x}(t), \end{cases} \tag{13.2}$$

where the input gain $\boldsymbol{g}(\boldsymbol{x})$ depends nonlinearly on the present state. Since the theory of PWA reconfiguration blocks developed in this monograph requires continuous vector fields, a constant approximation of the term $\boldsymbol{g}(\boldsymbol{x})$ is necessary. Therefore, the matrix $\boldsymbol{B}$ used in the PWA model (9.1) is taken as the Jacobian evaluated at the typical operating point of the process, as it was described in Chapter 4.1.

The states $x_{\mathrm{CW}}$, $x_{\mathrm{el,TS}}$, and $x_{\mathrm{el,TB}}$ of the thermofluid process are governed by linear dynamics, it is not necessary to approximate the corresponding part of the model. For this reason, the state was split into a linear part and a nonlinear part, which are in series interconnection, where the output of the linear part is the input of the nonlinear part. Only the state set corresponding to the nonlinear part was approximated by PWA dynamics. This part was split into 48 simplices that cover the cube 10 °C $\leq \vartheta_{\mathrm{TS}} \leq$ 40 °C, 0 m $\leq l_{\mathrm{TS}} \leq$ 0.4 m, 0.4 mS/cm $\leq \nu_{\mathrm{TS}} \leq$ 4.5 mS/cm, and 10 °C $\leq \vartheta_{\mathrm{TB}} \leq$ 40 °C. The entire PWA model was obtained by composing the linear part (three states) and the approximation of the nonlinear part (four states). Consequently, the overall state-space partition does not consist of simplices but of more general polytopes. The complete PWA model is given in Appendix E.2.

### *Nominal Controller*

A decentralised control scheme is applied to solve the control task for the nominal process. Three proportional-integral (PI) controllers

$$u_{\mathrm{TB}}(t) = k_P(r_{\nu,\mathrm{TS}}(t) - \nu_{\mathrm{TS}}(t)) + k_I \int_0^t (r_{\nu,\mathrm{TS}}(\tau) - \nu_{\mathrm{TS}}(\tau))\mathrm{d}\tau$$

$$u_{\mathrm{el,TS}}(t) = k_P(r_{\vartheta,\mathrm{TS}}(t) - \vartheta_{\mathrm{TS}}(t)) + k_I \int_0^t (r_{\vartheta,\mathrm{TS}}(\tau) - \vartheta_{\mathrm{TS}}(\tau))\mathrm{d}\tau$$

$$u_{\mathrm{PS}}(t) = k_P(r_{l,\mathrm{TS}}(t) - l_{\mathrm{TS}}(t)) + k_I \int_0^t (r_{l,\mathrm{TS}}(\tau) - l_{\mathrm{TS}}(\tau))\mathrm{d}\tau$$

with distinct proportional gains $k_P$ and integral gains $k_I$ are employed for the regulation of $\nu_{\mathrm{TS}}$, $l_{\mathrm{TS}}$ and $\vartheta_{\mathrm{TS}}$, where $r_{(\cdot)}$ denotes the respective reference signal. All controllers operate at a fixed sampling time of 200 ms.

The controller parameters given in Table 13.2 were obtained from initial tuning using the nonlinear process model, followed by fine-tuning on the physical process. The setpoint following requirement motivates the use of PI controllers, whereas the tuning aims at fast response with little overshoot. The design procedure reflects common practice in industry, where decentralised controllers are preferred over coupled controllers for ease of tuning and maintenance, whenever the process allows such control strategies.

**Table 13.2** Parameters of the nominal controllers.

| Loop | $k_P$ | $k_I$ |
|---|---|---|
| Electrical conductivity $\nu_{\mathrm{TS}}$ | 30 | 0.1 |
| Level $l_{\mathrm{TS}}$ | -25 | -0.15 |
| Temperature $\vartheta_{\mathrm{TS}}$ | 2 | 0.025 |

The behaviour of the nominal control loop after an increase of the setpoints for the temperature $\vartheta_{\mathrm{TS}}$ at $t = 300$ s, the fluid level $l_{\mathrm{TS}}$ at $t = 250$ s, and the electrical conductivity $\nu_{\mathrm{TS}}$ at $t = 200$ s is shown in Fig. 13.3. The figure shows a simulation, where a detailed nonlinear model is used to represent the process. The upper three axes show the process outputs, while the lower axes represent the inputs. The new setpoints are reached quickly by moving the heater $u_{\mathrm{el,TS}}$, the pump $u_{\mathrm{PS}}$, and the valve $u_{\mathrm{TB}}$. In this and in all following figures, grey boxes symbolise hard constraints on the fluid level imposed by the heating rods, which are forbidden to operate in dry condition, and by the reactor height.

## 13.3 Fault Scenarios and Redundancy

Several fault scenarios are used in the sequel to test the control reconfiguration methods by means of virtual sensors and virtual actuators. Actuator faults are denoted by $f_a$, whereas sensor faults are denoted by $f_s$.

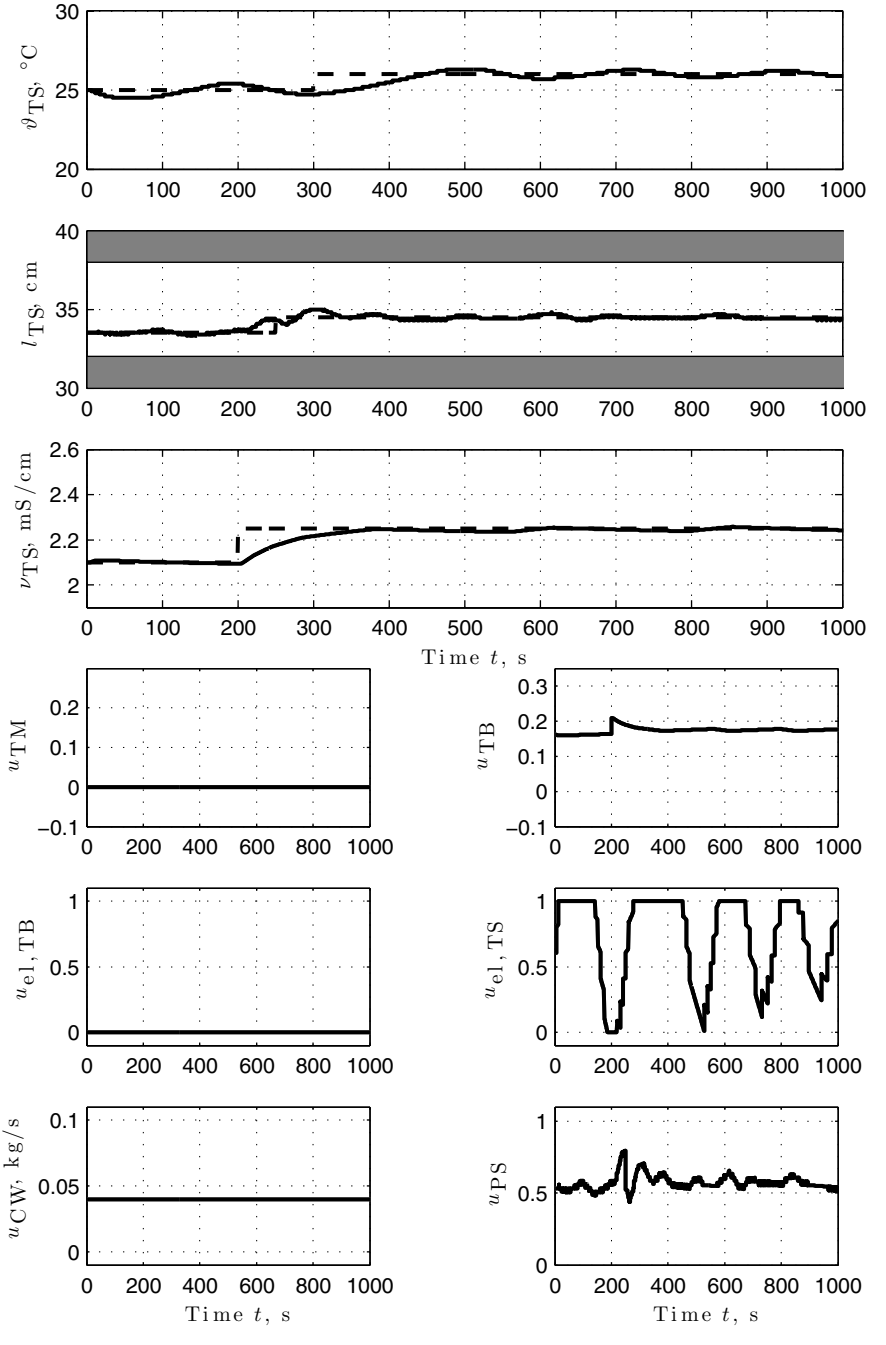


**Fig. 13.3** Nominal experimental process response to setpoint changes for temperature $\vartheta_{TS}$, level $l_{TS}$, and electrical conductivity $\nu_{TS}$.

### *Actuator Faults*

Each actuator fault represents a complete failure of at least one actuator, such that for any fault, at least one of the nominal control loops for $\nu_{\mathrm{TS}}$, $\vartheta_{\mathrm{TS}}$, or $l_{\mathrm{TS}}$ is broken. All actuator faults occur abruptly at time $t_f = 210$ s in all experiments and remain active thereafter (permanent, non-transient faults). To assure that the results are informative with respect to the reconfiguration success, each experiment involves a setpoint change that requires using the respective failed actuator.

The first actuator fault is a blockage of the valve $u_{\mathrm{TB}}$ at the equilibrium point, which is represented by the new input matrix:

$$f_{a1} : u_{\mathrm{TB}}(t > t_f) = \bar{u}_{\mathrm{TB}},\ \boldsymbol{B}_{f1} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{0} & \boldsymbol{b}_3 & \boldsymbol{b}_4 & \boldsymbol{b}_5 & \boldsymbol{b}_6 \end{pmatrix}.$$

The nominal loop for the electrical conductivity $\nu_{\mathrm{TS}}$ is broken, because the inflow rate to reactor TS from the reactor TB cannot be manipulated any more. This fault is realistic because most of the time, the valve operates near its equilibrium, unless disturbances have to be compensated. Furthermore, many technological valves have fail-safe positions for which the equilibrium values can be chosen.

The second actuator fault is a failure of the heater $u_{\mathrm{el,TS}}$ at the equilibrium, which is described by the following matrix:

$$f_{a2} : u_{\mathrm{el,TS}}(t > t_f) = \bar{u}_{\mathrm{el,TS}},\ \boldsymbol{B}_{f2} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \boldsymbol{b}_3 & \boldsymbol{0} & \boldsymbol{b}_5 & \boldsymbol{b}_6 \end{pmatrix}.$$

Since the temperature control loop uses the heater in the reactor TS, the control of the temperature $\vartheta_{\mathrm{TS}}$ is ineffective after the fault $f_2$ occurs. Failure at the operating point means that neither can further heater elements be switched on nor can operating ones be switched off. The latter scenario occurs when relays get stuck in closed position.

The third actuator fault is a failure of the pump speed control loop while the pump speed $u_{\mathrm{PS}}$ is at the equilibrium,

$$f_{a3} : u_{\mathrm{PS}}(t > t_f) = \bar{u}_{\mathrm{PS}},\ \boldsymbol{B}_{f3} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \boldsymbol{b}_3 & \boldsymbol{b}_4 & \boldsymbol{b}_5 & \boldsymbol{0} \end{pmatrix}.$$

The outflow from the reactor TS is fixed after this fault, and the nominal level control loop is broken. The technological representation of this fault is a failure of the pneumatic control valve which determines the pump velocity, while the pump runs at its equilibrium speed. All further actuator faults are combinations of the previous faults.

The fourth actuator fault is a combination of the heater failure $f_{a2}$ and the pump failure $f_{a3}$,

$$f_{a4} : u_{\mathrm{el,TS}}(t > t_f) = \bar{u}_{\mathrm{el,TS}},\ u_{\mathrm{PS}}(t > t_f) = \bar{u}_{\mathrm{PS}},\ \boldsymbol{B}_{f4} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{b}_2 & \boldsymbol{b}_3 & \boldsymbol{0} & \boldsymbol{b}_5 & \boldsymbol{0} \end{pmatrix}.$$

The fifth actuator fault is a combination of the valve failure $f_{a1}$ and the pump failure $f_{a3}$,

$$f_{a5} : u_{\mathrm{TB}}(t > t_f) = \bar{u}_{\mathrm{TB}},\ u_{\mathrm{PS}}(t > t_f) = \bar{u}_{\mathrm{PS}},\ \boldsymbol{B}_{f5} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{0} & \boldsymbol{b}_3 & \boldsymbol{b}_4 & \boldsymbol{b}_5 & \boldsymbol{0} \end{pmatrix}.$$

The sixth actuator fault is a combination of the valve failure $f_{a1}$ and the heater failure $f_{a2}$,

$$f_{a6} : u_{\mathrm{TB}}(t > t_f) = \bar{u}_{\mathrm{TB}},\ u_{\mathrm{el,TS}}(t > t_f) = \bar{u}_{\mathrm{el,TS}},\ \boldsymbol{B}_{f6} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{0} & \boldsymbol{b}_3 & \boldsymbol{0} & \boldsymbol{b}_5 & \boldsymbol{b}_6 \end{pmatrix}.$$

The seventh actuator fault is a combination of all three main actuator failures $f_{a1}$, $f_{a2}$, and $f_{a3}$,

$$f_{a7} : u_{\mathrm{TB}}(t > t_f) = \bar{u}_{\mathrm{TB}},\ u_{\mathrm{el,TS}}(t > t_f) = \bar{u}_{\mathrm{el,TS}},\ u_{\mathrm{PS}}(t > t_f) = \bar{u}_{\mathrm{PS}},$$
$$\boldsymbol{B}_{f7} = \begin{pmatrix} \boldsymbol{b}_1 & \boldsymbol{0} & \boldsymbol{b}_3 & \boldsymbol{0} & \boldsymbol{b}_5 & \boldsymbol{0} \end{pmatrix}.$$

The eighth and last actuator fault is a combined failure of both valve $u_{\mathrm{TB}}$ ($f_{a1}$) and the redundant valve $u_{\mathrm{TM}}$

$$f_{a8} : u_{\mathrm{TB}}(t > t_f) = \bar{u}_{\mathrm{TB}},\ u_{\mathrm{TM}}(t > t_f) = \bar{u}_{\mathrm{TM}},\ \boldsymbol{B}_{f8} = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{b}_3 & \boldsymbol{b}_4 & \boldsymbol{b}_5 & \boldsymbol{b}_6 \end{pmatrix}.$$

One actuator redundancy is the additional but not identical salt concentrate reservoir in the reactor TM with an electrical conductivity of $\nu_{\mathrm{TM}} = 4.5$ mS/cm and a temperature of $\vartheta_{\mathrm{TM}} = 24.5$ °C. The concentrate is accessible for TS via the control valve $u_{\mathrm{TM}}$, which remains closed during nominal process operation. Further available redundant actuation components are the heater with control input $u_{\mathrm{el,TB}}$ in the supply tank TB, and the cold water supply through valve $u_{\mathrm{CW}}$. In particular, the valve $u_{\mathrm{CW}}$ shows a marked nondeterministic quantised behaviour, hence it is best-suited for constant position operation.

### *Sensor Faults*

The considered sensor faults are single failures of sensor that measure the regulated variables, namely the fluid level $l_{\mathrm{TS}}$, the fluid temperature $\vartheta_{\mathrm{TS}}$, and the fluid electrical conductivity $\nu_{\mathrm{TS}}$.

The first sensor fault is a failure of the sensor for the temperature $\vartheta_{\mathrm{TS}}$,

$$f_{s1} : y_1(t > t_f) = 0,\ \boldsymbol{C}_{f1} = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{c}_2^T & \boldsymbol{c}_3^T & \boldsymbol{c}_4^T \end{pmatrix}^T,$$

the second sensor fault is a failure of the sensor for the level $l_{\mathrm{TS}}$,

$$f_{s2} : y_2(t > t_f) = 0,\ \boldsymbol{C}_{f2} = \begin{pmatrix} \boldsymbol{c}_1^T & \boldsymbol{0} & \boldsymbol{c}_3^T & \boldsymbol{c}_4^T \end{pmatrix}^T,$$

and the third sensor fault is a failure of the sensor for the level $\nu_{\mathrm{TS}}$,

$$f_{s3} : y_3(t > t_f) = 0,\ \boldsymbol{C}_{f3} = \begin{pmatrix} \boldsymbol{c}_1^T & \boldsymbol{c}_2^T & \boldsymbol{0} & \boldsymbol{c}_4^T \end{pmatrix}^T.$$

The fault diagnosis task is assumed to be perfectly solved throughout the study, as far as detection and isolation are concerned. The fault isolation event triggers

the model-based design of the virtual actuator and virtual sensor, respectively, to reconfigure the control law in response to the faults according to one of the reconfiguration algorithms. Subsequently, a virtual actuator or virtual sensor is activated by replacing its inactivity parameters with newly designed ones.

## 13.4 Reconfigurability Analysis

This section presents the results of a comprehensive reconfigurability analysis for the thermofluid process. Although the real plant exhibits nonlinear behaviour, it is helpful to include the linear conditions into a comprehensive reconfigurability analysis for two reasons.

First, they are necessary and sufficient, and therefore crisp conditions, and the violation of such a condition implies that the reconfiguration is not even possible locally around the chosen equilibrium. The practical system, which might deviate considerably from the equilibrium neighborhood, is unlikely to be easier to reconfigure than its linear model. In summary, the linear conditions are particularly informative when they provide negative answers to the question whether or not a reconfiguration problem is solvable. Second, linear conditions are easy to check, and they are readily available, usually in a numerically stable way.

However, if linear conditions are satisfied, this result bears limited information about the true plant with nonlinear dynamics and input constraints. For this reason, a thorough reconfigurability analysis also checks sufficient reconfigurability conditions obtained from the nonlinear reconfiguration approaches. In the case where they are not satisfied, it is unknown whether or not suitable reconfigured controllers exist, since the conditions are only sufficient.

### *Linear Reconfigurability Analysis*

**Actuator faults.** Table 13.3 shows the solvability of the linear reconfiguration problems for all defined actuator faults along with the used conditions for deciding the solvability of the problems. The table shows that stability can always be achieved, which is not surprising since the process is open-loop stable. The process also has sufficient redundancy so that the setpoint tracking property can be recovered after every fault.

However, the exact performance recovery goal can only be attained after fault $f_{a2}$, namely after heater failure. This fact has a clear physical interpretation. The heater control signal affects the fluid temperature in the reactor TS through first-order delay dynamics. The alternative ways to achieve the same effect consist in changing the cold-water inflow $u_{\mathrm{CW}}$, which also has first-order delay dynamics, as well as the heater $u_{\mathrm{el,TB}}$ in the tank TB, which likewise has first-order delay dynamics. There exists sufficient redundancy to independently influence temperature, level, and electrical conductivity with nominal performance. In all other fault scenarios, the actuators that fail have a more direct influence on some controlled variable than any possible replacement actuators. For example, the pump has a direct effect on fluid

**Table 13.3** Linear reconfigurability analysis for thermofluid process subject to actuator faults.

| Goal | Problem | Condition | $f_{a1}$ | $f_{a2}$ | $f_{a3}$ | $f_{a4}$ | $f_{a5}$ | $f_{a6}$ | $f_{a7}$ | $f_{a8}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Stability recovery | 4.1 | (4.53) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Setpoint tracking recovery | 4.2 | (4.54) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Exact performance recovery | 4.3 | (4.55) | × | ✓ | × | × | × | × | × | × |

Legend: ✓: solvable; ×: not solvable.

level, and a redundant level actuation consists in the combined actuation of cold water and salt concentrate inflows. The cold water supply is, however, governed by first-order delay dynamics.

Since exact performance recovery is not solvable in most fault cases, the optimal performance recovery is analysed. Figure 13.4 illustrates the solvability of optimal performance recovery (Problem 4.5). The figure shows numerical Pareto fronts on a logarithmic scale obtained from solving 100 different values for $\lambda \in [0, 1]$ for each fault case. The shown achievable $H_\infty$-norms have been computed using the LMI solver Sedumi 1.1. The convex curved fronts represent fault cases without numerical problems. Scattered clouds or non-convex curves indicate numerical problems.

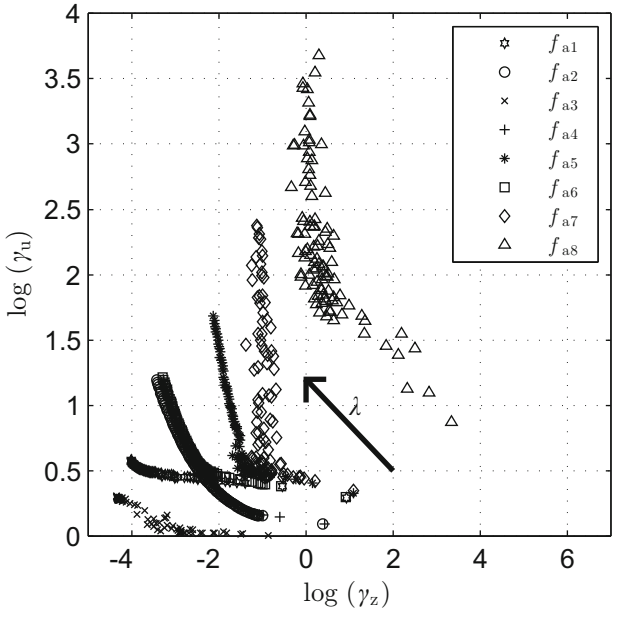


**Fig. 13.4** Achievable $H_\infty$-norms on input amplification and output error for linear model of thermofluid process subject to actuator faults.

In spite of the numerical problems, the diagram has an interesting interpretation in terms of reconfigurability analysis. Pareto fronts that lie in the lower-left corner of the diagram promise better reconfiguration success: they permit better output trajectory recovery at less input amplification than upper-right curves. The notation $f_k < f_l$ means that $f_k$ is easier to reconfigure than $f_l$. With this notation, it is easy to see from Fig. 13.4 that $f_{a3} < f_{a1} < f_{a2} < f_{a4} < f_{a6} < f_{a5} < f_{a7} < f_{a8}$. As intuitively expected, all single-actuator faults ($f_{a1}, f_{a2}, f_{a3}$) are easier to reconfigure than the two-actuator-faults ($f_{a4}, f_{a5}, f_{a6}, f_{a8}$). However, the triple-actuator fault $f_{a7}$ is easier to reconfigure than the two-actuator fault $f_{a8}$. The latter fault falls out of the fault pattern in the sense that the failing actuator $u_{\mathrm{TM}}$ is a redundant actuator, whereas all other faults only concern main actuators. Apparently, the failure of both valves that control salt concentrate inflow makes the control of electrical conductivity difficult.

The diagram has a further interesting interpretation relevant for the design of new plants. For some fault cases, the corresponding set of points is far spread out through the diagram, ranging over several orders of magnitudes. This means that the choice of the weight $\lambda$, being the free parameter of the reconfiguration algorithm, has strong influence on the reconfiguration result. For example, the set of points associated with fault $f_{a7}$ spans two orders of magnitude in $\gamma_u$ and one order of magnitude in $\gamma_z$. Thus, the parameter $\lambda$ has strong influence on both the output error and the required control effort, and a careful balancing of these contradicting goals is required. On the other hand, the fault $f_{a1}$ spans only a fraction of an order of magnitude in $\gamma_u$, but three orders of magnitude in $\gamma_z$, so it is reasonable to put more weight on the performance index $\gamma_z$ without considerable increase in control effort. In summary, the example shows that the weight $\lambda$ should depend on the fault case in practical applications.

**Sensor faults.** Table 13.4 shows the solvability of the linear reconfiguration problems for all defined sensor faults. Stable reconfiguration is always achievable, whereas tracking recovery in the presence of the disturbance on the electrical conductivity is only possible after the failure of the temperature sensor ($f_{s1}$) and after the failure of the level sensor ($f_{s2}$). After a failure of the sensor for the electrical conductivity ($f_{s3}$), the disturbance effect acting on the electrical conductivity cannot be detected and separated any longer, therefore the setpoint tracking recovery is not solvable. Exact performance recovery is only possible after failure of the level sensor ($f_{s2}$).

**Table 13.4** Linear reconfigurability analysis for thermofluid process subject to sensor faults.

| Goal | Problem | Condition | $f_{s1}$ | $f_{s2}$ | $f_{s3}$ |
|---|---|---|---|---|---|
| Stability | 4.1 | (4.31) | ✓ | ✓ | ✓ |
| Setpoint tracking | 4.2 | (4.32) | ✓ | ✓ | × |
| Exact performance recovery | 4.3 | (4.33) | × | ✓ | × |

Legend: ✓: solvable; ×: not solvable.

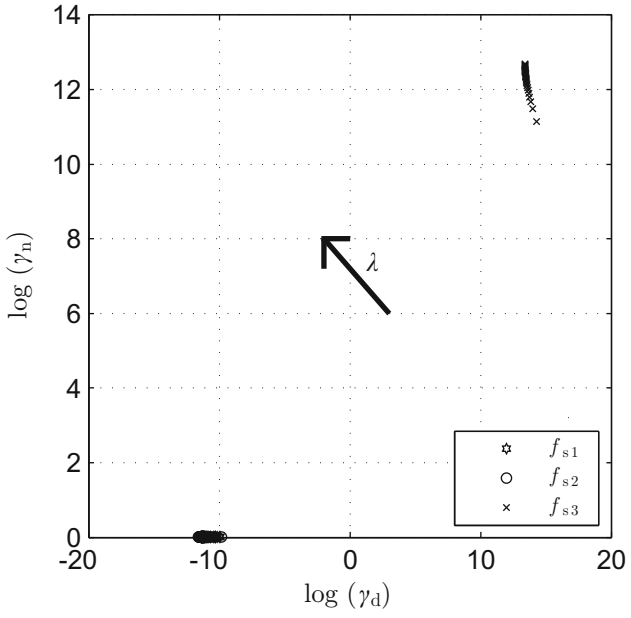


**Fig. 13.5** Achievable $H_\infty$-norms on input amplification and output error for linear model of thermofluid process subject to sensor faults.

Since exact performance recovery is only achievable in one sensor fault case, the solvability of optimal performance recovery is also investigated. The achievable $H_\infty$-norms for various weights $\lambda$ to solve Problem 4.5 according to Theorem 4.7 are shown in Fig. 13.5. The analysis shows that although only the fault case $f_{s2}$ permits exact performance recovery, the recovery can be much better approximated in the fault cases $f_{s1}$ as well as $f_{s2}$ than in the fault case $f_{s3}$. The fault case $f_{s3}$ turns out to be the most problematic sensor fault case with respect to the influence of disturbances and noise on the observation error. Taking into account the logarithmic scale, the fault case $f_{s3}$ can be considered to be not reconfigurable in practice.

The figure also shows that the achieved performance depends much less on the weight $\lambda$ than in the actuator case. Therefore, little attention has to be paid to the choice of the weight, in any fault case.

## *Hammerstein-Wiener Reconfigurability Analysis*

**Actuator faults.** Since the matrix $\boldsymbol{A}$ for the thermofluid process is Hurwitz, it is not surprising that the LMI (6.8) is feasible for every actuator fault case $f_{a1}$ through $f_{a8}$, and that the LMI (6.11) is feasible for every sensor fault case $f_{s1}$ through $f_{s3}$. Figure 13.6 shows the achievable solutions of Theorem 8.3 for varying weight $\lambda \in [0, 1]$. The figure shows similar characteristics as Fig. 13.4, up to certain differences in the position, spread and ordering of the point sets. The partial order of the actuator

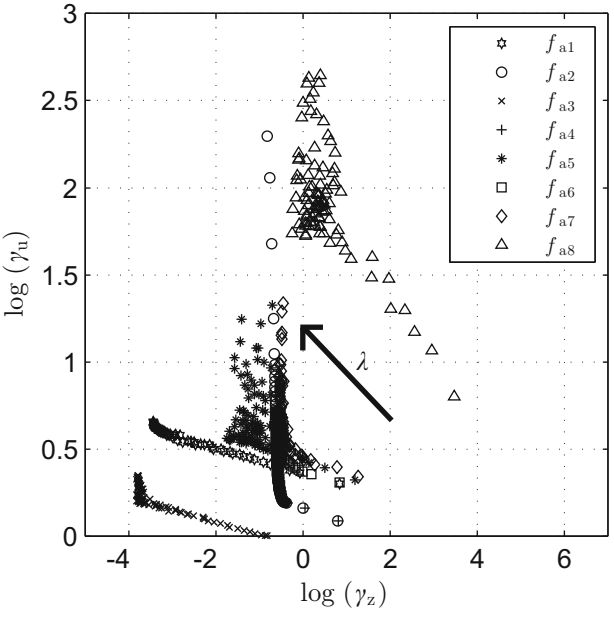


**Fig. 13.6** Achievable $H_\infty$-norms on input amplification and output error for linear subsystem of saturated thermofluid process subject to actuator faults.

fault reconfigurability is $f_{a3} < f_{a1} < f_{a2} < f_{a5} < f_{a4} < f_{a6} < f_{a7} < f_{a8}$. In particular, the single-fault cases $f_{a1}, f_{a2}, f_{a3}$ are still easier to reconfigure than the multiple actuator faults. In general, the achievable performance is worse than in the linear case (compare Fig. 13.6 to Fig. 13.4). This result is not surprising, since the solutions are restricted to maintaining stability in the presence of actuator saturations.

**Sensor faults.** The reconfigurability situation after sensor faults is the same as in the linear case, since no nonlinear output function is modelled in the thermofluid process. Therefore, the dynamics of the observation error is the same as in the linear case.

### *Piecewise Affine Reconfigurability Analysis*

**Actuator faults.** In the analysis of the reconfigurability after actuator faults, it is assumed that the state of the faulty plant is measurable, so that no virtual sensor is needed and the conditions describing its synthesis are ignored. In the case of piecewise affine systems and pure actuator faults, the reconfigurability analysis amounts to verifying the satisfiability of the LMIs (10.13) with regard to the stability recovery, and the LMIs (11.23) with regard to the stability and tracking recovery. Table 13.5 summarises the solvability with regard to stability recovery and tracking recovery for every fault case. Reconfigurability is given for every actuator fault scenario and for both goals.

**Table 13.5** Piecewise affine reconfigurability analysis for thermofluid process subject to actuator faults.

| Goal | Problem | Condition | $f_{a1}$ | $f_{a2}$ | $f_{a3}$ | $f_{a4}$ | $f_{a5}$ | $f_{a6}$ | $f_{a7}$ | $f_{a8}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| Stability recovery | 9.1 | (10.13) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Setpoint tracking recovery | 9.2 | (11.23) | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |

Legend: ✓: solvable; ×: not solvable.

## 13.5 Reconfiguration Applications

### *Hammerstein Virtual Actuator*

This section describes applications of the Hammerstein-Wiener virtual actuator presented in Chapters 5–8 to the reconfigurable control of the thermofluid process after actuator faults. The stability results of Chapter 6 are the basis of all simulations and experiments. In practice, the thermofluid process is very sensitive to fluid level deviations, since the band of admissible fluid levels is thin due to constraints given by heating rods and danger of overflow. This situation makes the reconfigurable control problem very hard to solve in practice. Of all synthesis methods for the saturated virtual actuator, the optimal performance recovery technique developed in Chapter 8 is best capable of successful reconfiguration such that the tight physical bounds of the plant are not exceeded in most fault cases. This result appears because the chosen synthesis method allows an explicit tradeoff between input amplification and output recovery, because it considers not only the equilibrium but also the transients, and because different weights can be attributed to the different outputs.

All forthcoming simulations are based on a detailed nonlinear model of the thermofluid process. All simulations and experiments are based on Algorithm 8.1 applied with differing tradeoff parameters $\lambda$ and differing weight matrices $\boldsymbol{Q}$ that define relative priorities between fluid level $l_{\mathrm{TS}}$, temperature $\vartheta_{\mathrm{TS}}$, and electrical conductivity $\nu_{\mathrm{TS}}$. The common setting $\lambda = 0.3$ and $\boldsymbol{Q} = \boldsymbol{I}_3$ provides stability and the satisfaction of the process constraints in the fault cases $f_{a1}, f_{a2}, f_{a3}, f_{a5}, f_{a7}$. In the fault cases $f_{a4}$ and $f_{a6}$ that include heater failure, the fluid level overflows the reactor. For this reason, and to improve performance in the other fault cases, adaptation of the synthesis parameters $\lambda$ and $\boldsymbol{Q}$ occurs in the below results. In order to limit the space required for the presentation of the results, only the experimental results for the actuator fault scenarios $f_{a1}$ and $f_{a5}$ are shown.

**Valve failure.** The simulated reconfigured closed-loop response after valve failure ($f_{a1}$) with a saturated virtual actuator (7.5) is shown in Fig. 13.7. The virtual actuator design uses multiobjective synthesis with $\lambda = 0.7$ and $\boldsymbol{Q} = \boldsymbol{I}_3$. The virtual actuator gains that are obtained from Algorithm 8.1 are

$$\boldsymbol{M} = 10^4 \begin{pmatrix} -0.0005 & -0.0019 & 1.7640 & -0.0001 & 0.0001 & -0.0003 & -0.0022 \\ 0 & 0 & 0.0066 & 0 & 0 & 0 & 0 \\ 0.0007 & 0.0065 & -2.0421 & 0.0006 & -0.0001 & 0.0022 & 0.0032 \\ 0.001 & 0.0066 & -3.2787 & 0.0004 & -0.0002 & 0.0016 & 0.0051 \\ -0.0005 & -0.0021 & 0.8675 & -0.0001 & 0.0001 & -0.0005 & -0.0015 \\ -0.0003 & -0.0020 & 0.9319 & -0.0001 & 0 & -0.0005 & -0.0012 \end{pmatrix}$$

$$\boldsymbol{N} = \begin{pmatrix} 0.6261 & 0.8247 & -0.0183 & 0.0029 & -0.0053 & 0.0853 \\ 0.8247 & -0.6783 & -0.0432 & 0.2164 & -0.0449 & -0.2104 \\ -0.0183 & -0.0432 & 0.7888 & 0.0129 & 0.0150 & 0.0371 \\ 0.0029 & 0.2164 & 0.0129 & 0.7560 & -0.0229 & 0.0253 \\ -0.0053 & -0.0449 & 0.0150 & -0.0229 & 0.8075 & -0.0005 \\ 0.0853 & -0.2104 & 0.0371 & 0.0253 & -0.0005 & 0.8016 \end{pmatrix}.$$

The upper three axes show the controlled outputs, which reach their setpoints. Furthemore, for all three output variables, the transient response shows little overshoot and transient dynamics that are adequate compared to the given dynamics of this process. The lower six axes show the control inputs $\boldsymbol{u}_f$ acting on the faulty plant in black and the control inputs $\boldsymbol{u}_c$ given by the nominal controller in grey. Thus, the influence of the virtual actuator is clearly visible. The heater $u_{\mathrm{el,TS}}$ repeatedly reaches its saturation bounds, nevertheless the process is stable. The other inputs remain well within their given saturation bounds, which is a consequence of the good damping imposed by the passivity condition that is embedded into the synthesis. The experimental results obtained with the saturated virtual actuator synthesised based on the same values for the design parameters $\lambda$ and $\boldsymbol{Q}$ are shown in Fig. 13.8. The experiment qualitatively confirms the observations made about the simulations. From now on, only experiments are shown where available.

**Valve and pump failure.** The experimental reconfigured closed-loop response after valve and pump failure ($f_{a5}$) using a saturated virtual actuator (7.5) is shown in Fig. 13.9. The virtual actuator design uses multiobjective synthesis with $\lambda = 0.5$ and $\boldsymbol{Q} = \mathrm{diag}(0.1\ 1\ 0.1)$. The virtual actuator gains that are obtained with this from Algorithm 8.1 are

$$\boldsymbol{M} = 10^3 \begin{pmatrix} 0 & 0.0067 & -0.0110 & 0.0001 & 0 & 0.0003 & -0.0001 \\ 0 & 0.0002 & -0.0884 & 0 & 0 & 0 & 0 \\ 0 & 0.0048 & -0.7839 & 0.0002 & 0 & 0.0007 & 0.0002 \\ 0.0001 & 0.0008 & -0.4092 & 0 & 0 & 0.0001 & 0.0002 \\ -0.0001 & 0.0044 & -1.0497 & 0.0001 & 0 & 0.0006 & 0.0003 \\ 0 & 0.0003 & -0.0944 & 0 & 0 & 0 & 0.0001 \end{pmatrix}$$

$$\boldsymbol{N} = 10^{-6} \begin{pmatrix} 0.4701 & 0.9156 & -0.0368 & -0.0497 & -0.0563 & -0.2101 \\ 0.9156 & -0.6128 & 0.1670 & -0.0651 & -0.0268 & 0.0931 \\ -0.0368 & 0.1670 & 0.8644 & -0.0826 & -0.0824 & 0.0121 \\ -0.0497 & -0.0651 & -0.0826 & 0.7925 & -0.0600 & -0.5281 \\ -0.0563 & -0.0268 & -0.0824 & -0.0600 & 0.7833 & -0.3012 \\ -0.2101 & 0.0931 & 0.0121 & -0.5281 & -0.3012 & -0.2285 \end{pmatrix}.$$

The controlled outputs shown in the three upper axes are all stable. The temperature and fluid level attain their setpoints, whereas the electrical conductivity exceeds its

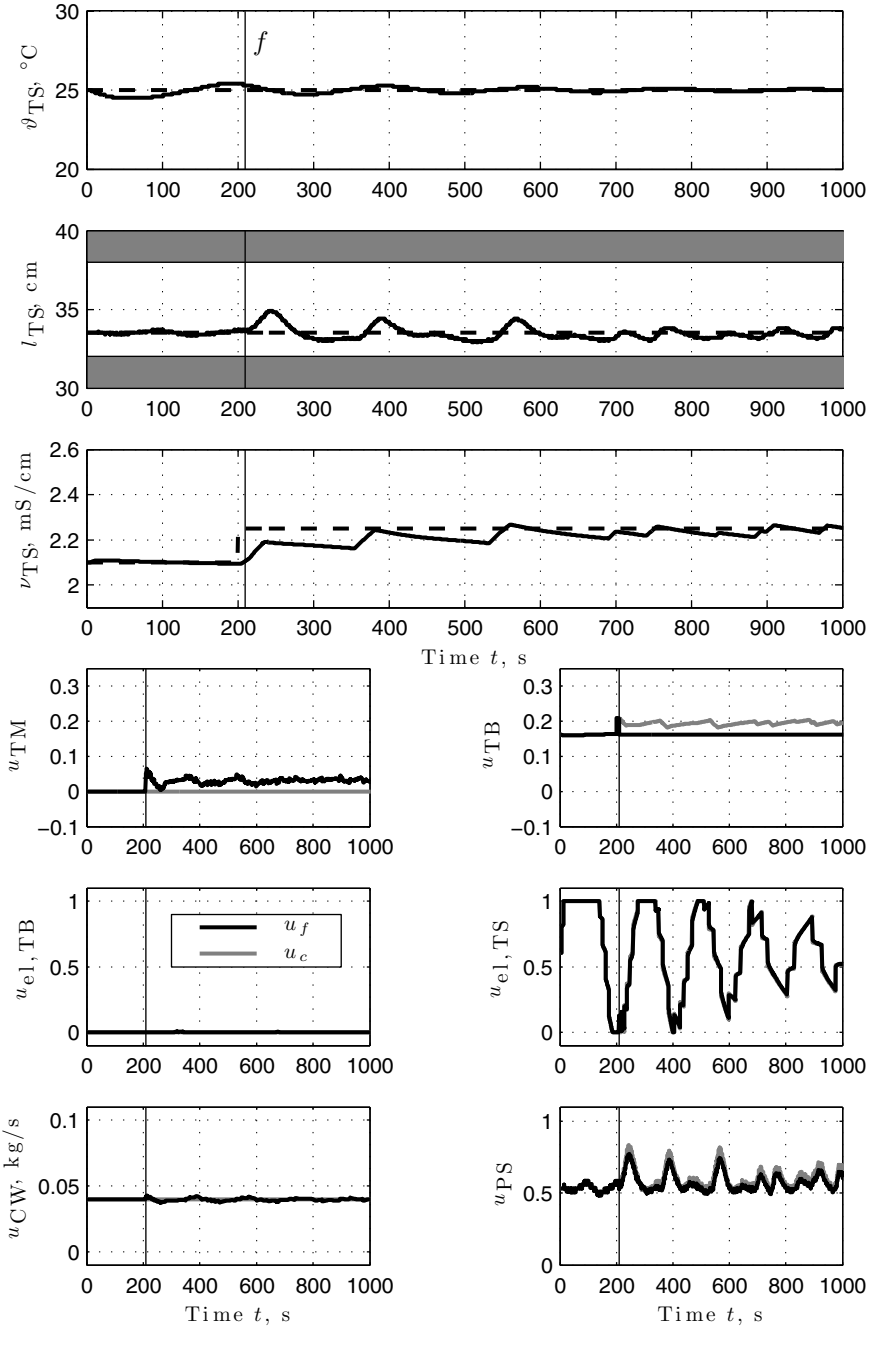


**Fig. 13.7** Simulated reconfigured closed-loop response after valve failure ($f_{a1}$) with saturated virtual actuator using multi-objective synthesis.

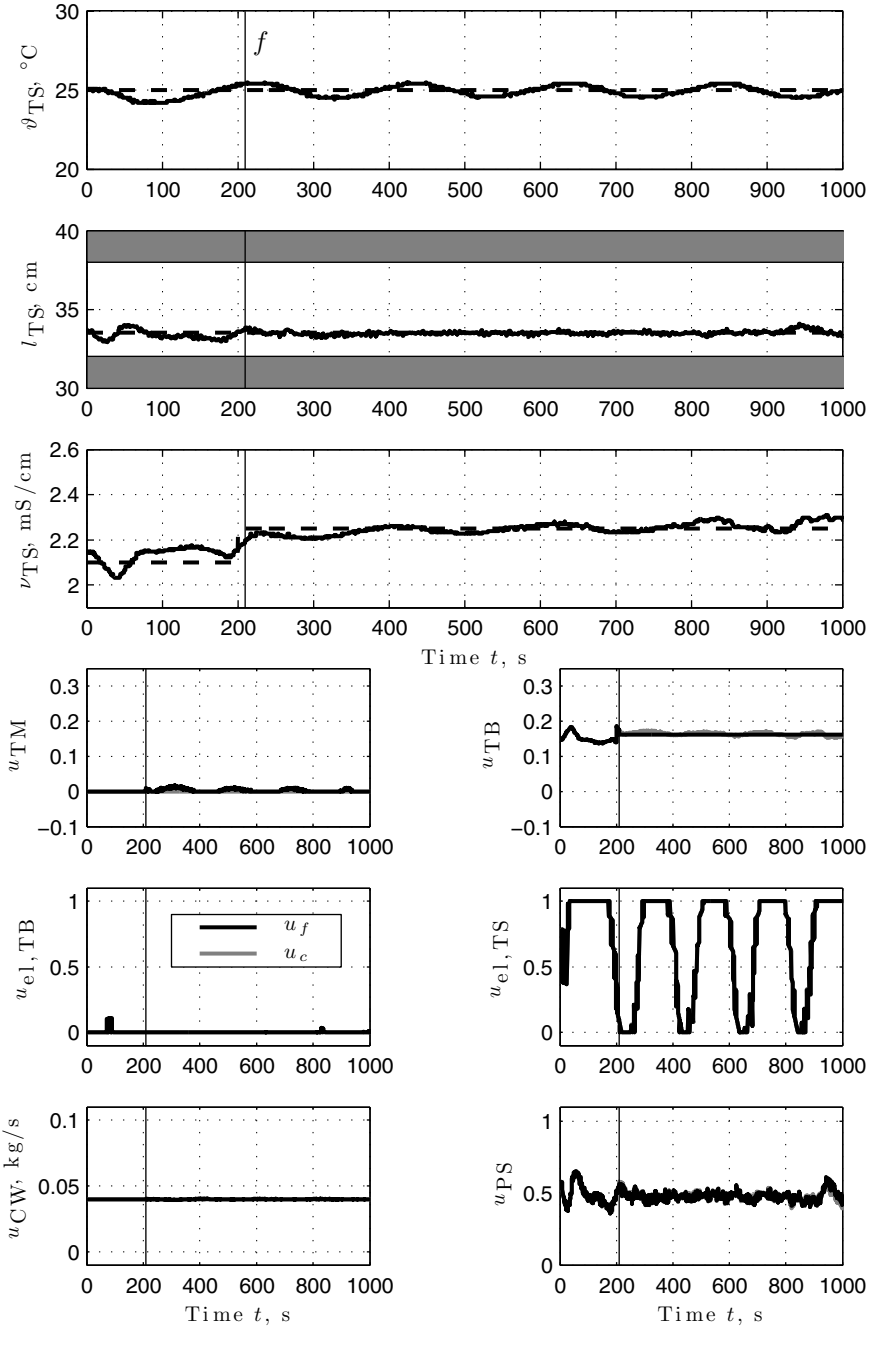


**Fig. 13.8** Experimental reconfigured closed-loop response after valve failure ($f_{a1}$) with saturated virtual actuator using multi-objective synthesis.

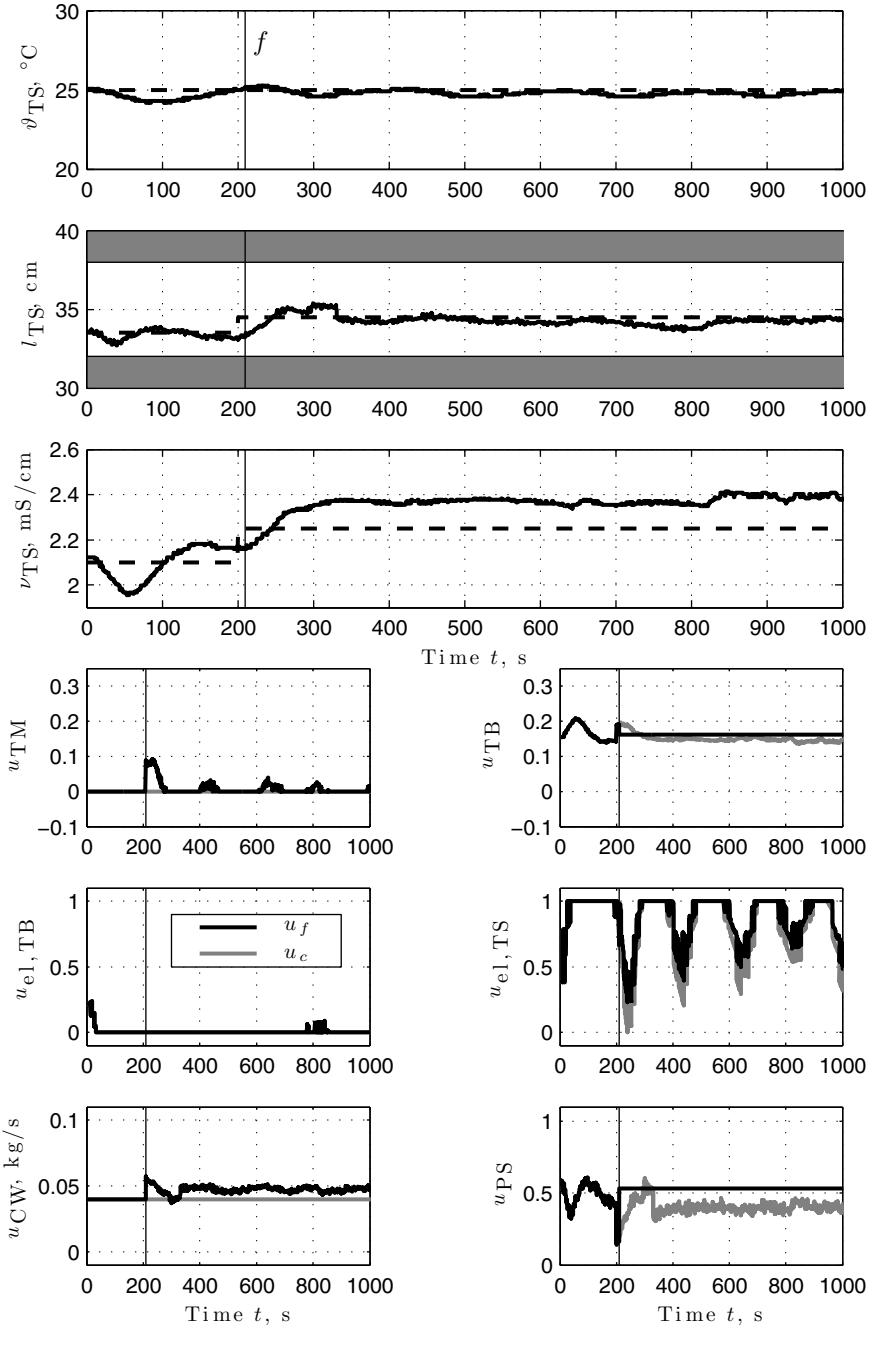


**Fig. 13.9** Experimental reconfigured closed-loop response after valve and pump failure ($f_{a5}$) with saturated virtual actuator using multi-objective synthesis.

setpoint. The transient response is smooth and well-damped. The experiment shows that the virtual actuator is sufficiently robust for stabilising the process in spite of unmodelled nonlinear dynamics. The only control input that reaches its saturation bounds is the electrical heater $u_{\mathrm{el,TS}}$. All other control inputs show small variations over the used time interval.

### *Hammerstein-Wiener Virtual Sensor*

This section describes applications of the saturated virtual sensor presented in Chapter 6 to the reconfigurable control of the thermofluid process after sensor faults. The stability results of Chapter 6 are the basis of all simulations and experiments. All simulations use the detailed nonlinear model to represent the process.

**Temperature sensor failure.** The experimental reconfigured closed-loop response after the failure of the temperature sensor in the reactor TS ($f_{s1}$) is shown in Fig. 13.10. The virtual sensor design based on multiobjective synthesis with $\lambda = 0.5$ results in the output error injection gain

$$\boldsymbol{L} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0.13 & 0 & 0 \\ 0 & 0 & 14.27 & 0 \\ 0 & 0 & 0 & 3.33 \\ 0 & -0.0014 & 0 & 0 \\ 0 & 0 & 0 & -0.38 \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

The virtual sensor stabilises the closed-loop system, as the solvability conditions promise. The state estimate for the temperature $\vartheta_{\mathrm{TS}}$ is biased by about 1 K. Altogether, the process remains stable, the fluid level $l_{\mathrm{TS}}$ and the electrical conductivity $\nu_{\mathrm{TS}}$ attain their setpoints, and the temperature stays very close to its setpoint. The estimation offset is due to model errors that enter the estimation error as disturbances.

**Level sensor failure.** The simulated reconfigured closed-loop response after the failure of the level sensor in the reactor TS ($f_{s2}$) is shown in Fig. 13.11. The virtual sensor design uses multiobjective synthesis with $\lambda = 0.3$ and results in the output error injection gain

$$\boldsymbol{L} = \begin{pmatrix} 0.87 & 0 & 0 & 0.41 \\ 0 & 0 & & 0 \quad 0 \\ 0 & 0 & 14.3 & 0 \\ 0.4 & 0 & 0 & 3.13 \\ 0.06 & 0 & 0 & 0.002 \\ -0.03 & 0 & 0 & -0.37 \\ -0.09 & 0 & 0 & -0.08 \end{pmatrix}.$$

The saturated virtual sensor stabilises the closed-loop system, as the solvability conditions promise. The fluid level is, however, estimated with a slight steady-state offset of about 2 cm. Still, the estimate is sufficiently accurate to keep the true outputs sufficiently close to their setpoints.

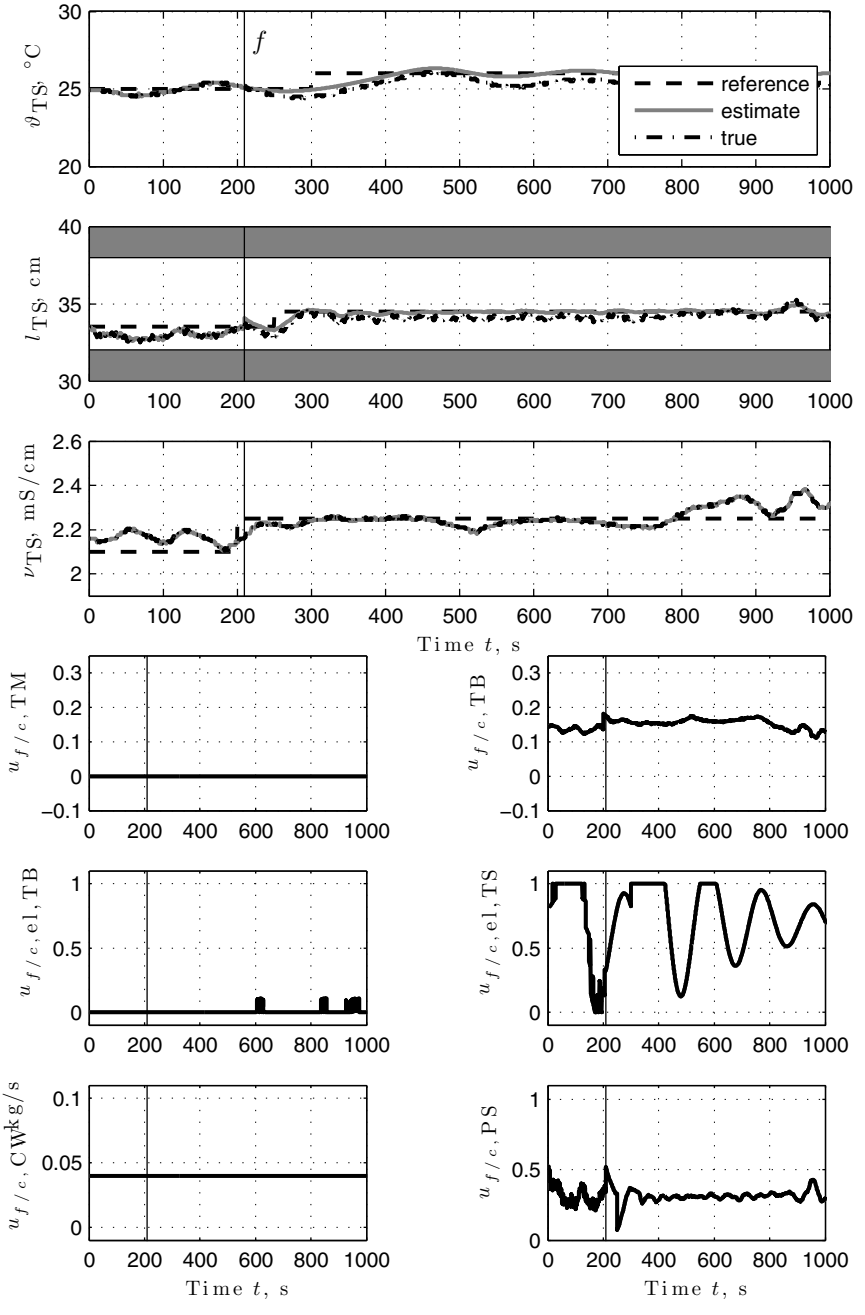


**Fig. 13.10** Experimental reconfigured closed-loop response after failure of temperature sensor for reactor TS ($f_{s1}$) with saturated virtual sensor using multi-objective synthesis.

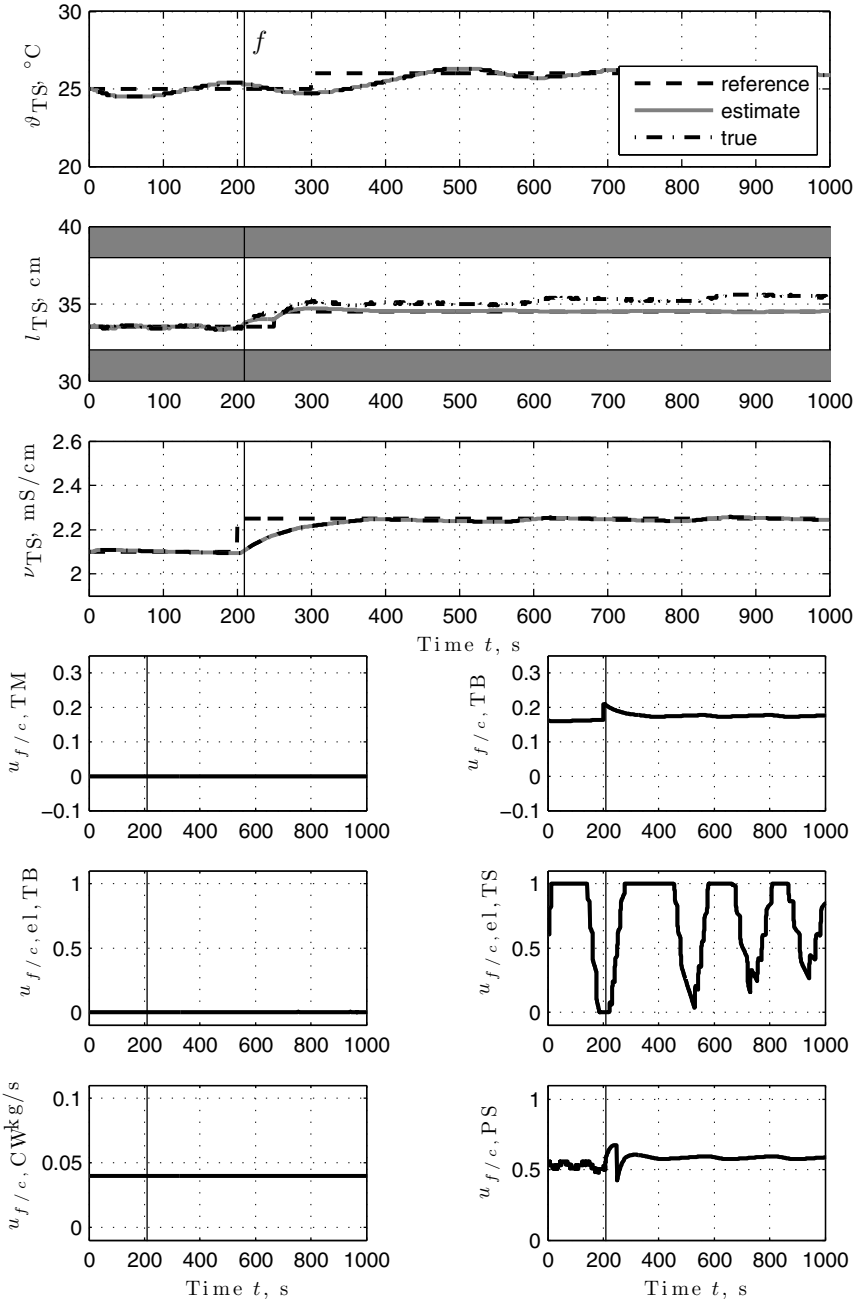


**Fig. 13.11** Simulated reconfigured closed-loop response after failure of level sensor for reactor TS ($f_{s2}$) with saturated virtual sensor using multi-objective synthesis.

### *Piecewise Affine Virtual Actuator*

**Valve failure.** The application of the PWA virtual actuator to the failure of the valve $u_{\mathrm{TB}}$ that governs the inflow of salt concentration from the container TB is shown as a simulation in Fig. 13.12. The simulation uses the detailed nonlinear model to represent the process. In variation of the original definition of the fault $f_{a1}$, the valve fails in completely closed position. The blockage compensation condition (9.9) is not satisfied. Although the valve $u_{\mathrm{TM}}$ opens the way to fluid with the same salt concentration in the container TM, that fluid also has a slightly different effect on the temperature $\vartheta_{\mathrm{TS}}$ than the fluid in the container TB. Obviously, the cold water inflow $u_{\mathrm{CW}}$ also affects the temperature. In summary, the blockage of the valve $u_{\mathrm{TB}}$ cannot be directly compensated by means of an addition to the affine terms.

Nevertheless, the upper three axes of Fig. 13.12 show that the PWA virtual actuator achieves satisfactory reconfiguration in the sense that all three controlled outputs are stable, the temperature and electrical conductivity reach their setpoints with good transients. Only the fluid level repeatedly reaches its lower safety bound and its mean value deviates slightly from its setpoint. In other words, stable setpoint tracking is achieved, and sufficient performance is achieved so that the transient response exhibits acceptable overshoot and oscillations. Indeed, the PWA virtual actuator undergoes switching between several different modes.

The closed-loop system that consists of the faulty plant, the virtual actuator, and the nominal controller has been modified with respect to Fig. 11.1 as follows. First, the real process is subjected to constraints on the inputs. Although these constraints are not part of the framework developed for PWA virtual actuators, it is advantageous in practice to constrain the control inputs $\boldsymbol{u}_f$ that are outputs of the PWA virtual actuator. Furthermore, the PWA virtual sensor was not used, since all process states were assumed to be measurable. The computation procedure for the gains $\boldsymbol{M}$ and $\boldsymbol{M}_I$ consists of Algorithm 11.1 extended by means of performance elements in the form of an upper bound on the $H_\infty$-norm of the PWA virtual actuator from $\boldsymbol{u}_c$ to $\boldsymbol{u}_f$ and from $\boldsymbol{u}_c$ to $\boldsymbol{z}_\Delta$. With these technique and the weights $\lambda = 0.7$ and $\boldsymbol{Q} = \mathrm{diag}(0.1\ 1\ 0.1)$, the gains

$$\boldsymbol{M} = \begin{pmatrix} 0.0547 & 22.2039 & 250.1074 & 0.0544 & 0.0915 & 0.2414 & 0.1422 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.0003 & 0.4615 & -49.6348 & 0.0292 & -0.0007 & 0.0787 & 0.0209 \\ -0.0124 & -1.7227 & -46.0860 & -0.0152 & -0.0009 & -0.0226 & 0.0054 \\ 0.0109 & 1.1141 & -225.2465 & 0.0333 & 0.0194 & 0.1280 & 0.0368 \\ -0.0192 & -2.2908 & 363.4965 & -0.0247 & -0.0231 & -0.0814 & -0.0390 \end{pmatrix}$$

$$\boldsymbol{M}_I = 10^{-6} \begin{pmatrix} 0.1121 & 0.0015 & -0.0127 \\ 0 & 0 & 0 \\ 0.1163 & 0.0002 & -0.0101 \\ -0.0154 & -0.0002 & 0.0011 \\ 0.0371 & -0.0002 & -0.0042 \\ -0.0358 & 0.0006 & 0.0049 \end{pmatrix}$$

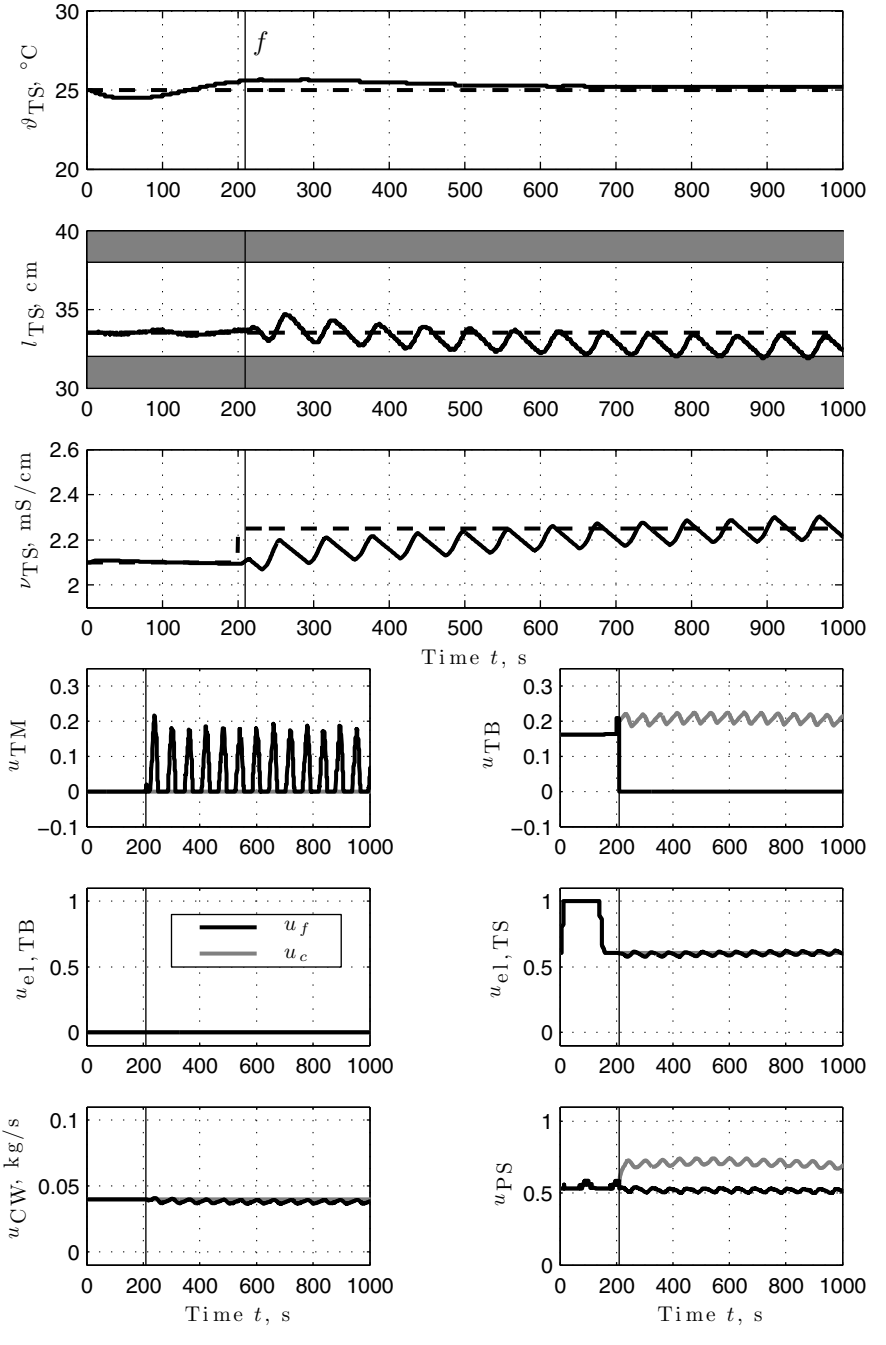


**Fig. 13.12** Simulated reconfigured closed-loop response after valve blockage in closed position (modified $f_{a1}$) with piecewise affine virtual actuator using tracking-recovering synthesis.

are obtained. The effect of these feedback gains is visible in the lower six axes of Fig. 13.12. The blockage position of the valve $u_{\mathrm{TB}}$ is clearly visible. Apparently, the valve $u_{\mathrm{TM}}$ to the spare salt container serves as the main replacement, with contributions of the cold water supply valve $u_{\mathrm{CW}}$. The pump rate is adjusted to fit the reduced throughput that results from the fault and reconfiguration.

In summary, the simulation demonstrates the adequateness of the PWA virtual actuator for reconfiguring systems described by PWA or approximately PWA dynamics. Further improvements of performance are likely to be achievable by further investigating the application of optimal control theory for PWA systems to PWA reconfiguration blocks [165].

## 13.6 Summary and Discussion

This chaper has summarised a selection of simulations and experiments of successful applications of saturated virtual actuators, saturated virtual sensors, and PWA virtual actuators to a thermofluid process. The shown sample applications demonstrate that practical applications of the fault-hiding paradigm to fault-tolerant control problems are feasible. In particular, successfully reconfiguration has been achieved in fault cases that cannot be solved based on linear virtual actuators.

Especially in the case of the saturated virtual actuator, additional simulations and successful experiments for further fault cases have been achieved. Reconfiguring additional sensor fault cases is difficult, since the failure of the fluid temperature and fluid level sensors prove to be sufficiently difficult both in terms of reconfigurability analysis and in terms of experiments.

On the other hand, the PWA virtual actuator and the PWA virtual sensor have so far only been tested in simulations. Their experimental evaluation on the thermoluid process is an open aspect for further investigation.

# Chapter 14
# Conclusion

**Abstract.** This chapter summarises the contributions of this monograph to reconfigurable control theory and describes open problems for future research.

## 14.1 Summary

The goal formulated in the introduction consisted in extending the linear fault-hiding approach to reconfigurable control towards two classes of nonlinear systems, namely towards Hammerstein-Wiener systems and piecewise affine systems with more than two modes. The reconfiguration solutions were supposed to address combined actuator and sensor faults, to encompass stability, tracking and performance properties, to be found by autonomous algorithms, and to be robust against uncertain diagnostic results. These goals have been achieved.

Part I of this monograph has defined and explained the reconfiguration problem in terms of general nonlinear system models. The general structure of fault-hiding solutions was defined as being a combination of a nonlinear virtual sensor for countering sensor faults and a nonlinear virtual actuator for countering actuator faults. Key properties of the fault-hiding solution have been stated independently of any particular subclass of nonlinear systems. The problems of stability recovery, setpoint tracking recovery, exact performance recovery, and optimal performance recovery were defined, and their solutions for linear systems were summarised. It was noted that the linear reconfiguration blocks are universal solutions to the linear reconfiguration problems.

Parts II and III have described the extensions to Hammerstein-Wiener systems and piecewise affine systems, that have been indeed completely achieved. The contributions consist in Hammerstein-Wiener and piecewise affine generalisations of virtual sensors and virtual actuators, as well as solutions to the stability recovery and setpoint recovery problems. A solution to the optimal performance recovery problem has been provided for the practically particularly relevant class of saturated systems. For each problem, sufficient solvability conditions are available that permit a reconfigurability analysis. In particular, the Hammerstein-Wiener framework

J.H. Richter: Reconfigurable Control of Nonlinear Dynamical Systems, LNCIS 408, pp. 229–232.
springerlink.com 

explicity includes the presence of input saturations. The price paid for the extension to nonlinear systems consists in the non-necessity of the obtained solvability conditions, since the solvability analysis leads to decision problems that are known to be generally undecidable from a computational perspective. Robustness against modelling errors has been found and demonstrated by means of running examples in all cases. It has been emphasised that the solutions are well-suited for autonomous implementation. Therefore, all specific requirements that apply to reconfigurable control problems have been met by the solutions. By virtue of algorithms and examples, it has been explained in each chapter how the autonomous implementation of the reconfiguration blocks can be achieved. The roles of remaining degrees of freedom for parameterising the algorithms for specific engineering applications have been explained and interpreted.

Part IV has provided implementation details for the fault-hiding approach and described an experimental application example. It has been shown how the fault-hiding principle can be embedded into practical real-time control frameworks, and a prototype toolbox implementation based on MATLAB and Simulink has been described. An experimental example from the process control domain applied to a pilot plant has demonstrated the viability of the approaches for solving practical fault-tolerant control problems in engineering applications. The experiments stress the robustness of the fault-hiding approach. Potential generalised applications to the control of underactuated systems have been indicated.

In summary, the problem of reconfiguring feedback control laws after actuator faults and sensor faults has been extensively and successfully solved in this monograph. A suitable generalisation of the linear fault-hiding framework towards nonlinear dynamical systems has been found. This generalisation has been worked out for Hammerstein-Wiener systems and piecewise affine systems. The class of admissible faults has been extended, most importantly from actuator failure at the operating point towards blockage of actuators in arbitrary positions (piecewise affine models), as well as towards tightened actuator saturation limits (Hammerstein-Wiener models). Nevertheless, several questions remain open, and new interesting problems have appeared, which are explained in the next section.

## 14.2 Open Problems

This monograph has entirely focussed on actuator and sensor faults that affect the input and output matrices of the models. However, certain classes of practical faults also affect the autonomous part of the plant dynamics. The extension of the fault-hiding approach towards internal system faults modelled by means of modified system matrices corresponding to the autonomous dynamics is an interesting open problem for all modelling approaches considered in this monograph.

The fault-hiding framework has so far been developed only for continuous-time systems. However, most control applications nowadays require a digital implementation that leads to sampled-data systems. This problem can be avoided by means of suitable oversampling, but this solution is limited to systems with sufficiently

slow dynamics. The extension of the fault-hiding framework towards sampled-data systems is a problem of high practical value. It is nontrivial even in the linear case, since in discrete-time linear systems, phenomena like dead-beat control appear that are not present in continuous-time systems. In the linear case, many discrete-time problems are expected to be solvable along the lines of the continuous-time solutions. The extension is certainly nontrivial in the nonlinear case, since no general approach to the treatment of sampled-data nonlinear systems is available.

In the case of piecewise affine systems, the replacement of common quadratic Lyapunov functions with piecewise quadratic or polyhedral Lyapunov functions is an intriguing but difficult problem. As discussed in Chapter 10.3, the observation error associated with the piecewise affine virtual sensor must be stable for every discrete mode combination in the plant and the virtual sensor, and gain scheduling based on piecewise quadratics defined on the observer error is not feasible. However, it might be possible to develop a piecewise Lyapunov function approach to virtual actuator synthesis for the case where the plant state is measurable and where no sensor faults occur. In addition, the optimal performance recovery problem has not been considered yet for piecewise affine virtual actuators and sensors.

Regarding nonlinear systems, the general nonlinear virtual actuators and virtual sensors defined in Chapter 3 can be further generalised by allowing nonlinear feedback gains. Furthermore, actuator constraints have not been taken into account in the fault-tolerant control of nonlinear dynamical systems. It would be interesting to develop synthesis techniques for generalised nonlinear virtual actuators and virtual sensors that respect constraints on the inputs, and possibly also on the states. Such an approach might be developed for the class of input-affine systems based on control Lyapunov functions, and also for further special subclasses of nonlinear systems, such as Lure systems, bilinear systems, and linear parameter-varying systems.

Furthermore, the reconfiguration framework should be generalised to two-degree-of-freedom control schemes. In two-degree-of-freedom control, an open-loop feedforward controller shapes the dynamics of the reference-to-output behaviour, while a feedback controller shapes the disturbance-to-output behaviour and provides robustness against modelling uncertainties. The feedforward controllers are, for example, based on the differential flatness property [58]. It has been shown that a flatness-based feedforward controller is robust against model uncertainties and that the flatness-based control scheme can linearise the system around its trajectory [72]. These properties opens interesting ways for combinations of linear reconfiguration blocks with differentially flat nonlinear systems. The reconfiguration of feedforward controllers has not been thoroughly studied yet, but could lead to very interesting general reconfiguration approaches for two-degree-of-freedom control schemes.

Taking a bigger perspective, industry moves towards ubiquitous and complete networking of all factory assets, such as actuators or sensors, by means of digital fieldbus networks integrated with corporate networks. This large-scale networking of components allows controllers to communicate with non-local and local sensors and actuators alike, as well as with other controllers and remote components. While automatic control over large-scale networks poses challenges that have led to the

field of networked control systems [76, 78, 217], it also offers unique opportunities to fault-tolerant control. However, systematic methods have yet to be developed that permit controllers to discover the assets available in its neighborhood, to asses their capabilities, and to use them for achieving their purposes. This reasoning applies especially if the global model of the entire neighborhood of a plant component is not immediately available, or not known at all. In such cases, local reconfiguration blocks should attempt globally sensible decisions based on local information, which is a typical decentralised control problem. Similar problems also arise in hierarchical and mixed decentralised/hierarchical control schemes. Their solutions require new theoretical methods as well as suitable modelling tools. The existing theory of decentralised control might provide a starting point.

The unified treatment of fault diagnosis and control adjustment is an area of research that is still in its infancy. This research area is concerned with the development of control adjustment techniques that take explicitly into account the uncertainties that are commonly associated with fault diagnosis results. Furthermore, the fault diagnosis task becomes more complicated in the presence of control adjustments. It is possible to combine the fault-hiding approach to control reconfiguration with the set-theoretic ideas formulated in [146] in order to obtain a fully integrated active fault-tolerant control scheme. This concept is explored for linear systems in [195].

Finally, two general questions have appeared in this monograph that are interesting beyond fault-tolerant control applications. The first question concerns conditions about the solvability of decoupling problems for general switched linear or switched affine systems. The second question concerns the characterisation of incremental stability conditions for nonlinear systems in terms of piecewise quadratic or polyhedral Lyapunov functions. These problems are of general interest, they are very difficult, and contributions to these questions are certainly very valuable.

# References


[1] Adelt, S.: Rekonfigurierbare Regelung stückweise affiner Systeme: Modellbildung und Experimente. Master thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2009); ATP 0064

[2] Ahmed-Zaid, F., Ioannou, P., Gousman, K., Rooney, R.: Accommodation of failures in the F-16 aircraft using adaptive control. IEEE Control Systems Magazine 11(1), 73–78 (1991)

[3] Ahvenlampi, T., Rantanen, R., Tervaskanto, M.: Fault tolerant control application for continuous kraft pulping process. In: Proc. 6th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes, Beijing, China, pp. 1513–1524 (2006)

[4] Alessandri, A., Coletta, P.: Design of observers for a class of hybrid linear systems. In: Di Benedetto, M.D., Sangiovanni-Vincentelli, A.L. (eds.) HSCC 2001. LNCS, vol. 2034, pp. 7–17. Springer, Heidelberg (2001)

[5] Angeli, D.: A Lyapunov approach to incremental stability properties. IEEE Trans. Autom. Control 47(3), 410–421 (2002)

[6] Aravena, J., Zhou, K., Li, X.R., Chowdhury, F.: Fault tolerant safe flight controller bank. In: Proceedings of the 6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (Safeprocess), Beijing, China, pp. 859–864 (2006)

[7] Ashari, A.E., Sedigh, A.K., Yazdanpanah, M.J.: Reconfigurable control system design using eigenstructure assignment: static, dynamic and robust approaches. Int. J. Control 78(13), 1005–1016 (2005)

[8] Askounis, D.T., Assimakopoulos, V., Psarras, J.: Fault tolerance in supervisory control systems: a knowledge-based approach. J. Intelligent Manufacturing 5(5), 323–331 (1994)

[9] Aßfalg, J., Allgöwer, F.: Fault diagnosis with structured augmented state models: modeling, analysis, and design. In: Proc. 2006 Conf. Decision Control, San Diego, USA, pp. 1165–1170 (2006)

[10] Balluchi, A., Benvenuti, L., di Benedetto, M., Sangiovanni-Vincentelli, A.: Design of observers for a class of hybrid linear systems. In: Tomlin, C.J., Greenstreet, M.R. (eds.) HSCC 2002. LNCS, vol. 2289, pp. 76–89. Springer, Heidelberg (2002)

[11] Baric, M., Grieder, P., Baotic, M., Morari, M.: An efficient algorithm for optimal control of PWA systems with polyhedral performance indices. Automatica 44(1), 296–301 (2008)

[12] Basile, G., Marro, G.: Controlled and Conditioned Invariants in Linear Systems Theory. Prentice-Hall, University of Bologna (1992), http://www3.deis.unibo.it/Staff/FullProf/GiovanniMarro/gm_books.htm (Out of print)
[13] Bemporad, A., Ferrari-Trecate, G., Morari, M.: Observability and controllability of piecewise affine and hybrid systems. IEEE Trans. Automatic Control 45(10), 1864–1876 (2000)
[14] Bemporad, A., Garulli, A., Paoletti, S., Vicino, A.: A bounded-error approach to piecewise affine system identification. IEEE Trans. Automatic Control 50(10), 1567–1580 (2005)
[15] Bemporad, A., Torrisi, F.D., Morari, M.: Optimization-based verification and stability characterization of piecewise affine and hybrid systems. In: Lynch, N.A., Krogh, B.H. (eds.) HSCC 2000. LNCS, vol. 1790, pp. 45–58. Springer, Heidelberg (2000)
[16] Ben-Israel, A., Greville, T.N.E.: Generalized Inverses, 2nd edn. CMS Books in Mathematics. Springer, Heidelberg (2003)
[17] Benosman, M., Lum, K.Y.: On-line references reshaping and control reconfiguration for non-minimum phase nonlinear fault-tolerant control. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 2563–2569 (2008)
[18] Benosman, M., Lum, K.Y.: Passive actuators' fault tolerant control for affine nonlinear systems. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 14, 229–234 (2008)
[19] Bernstein, D.: Matrix Mathematics. Princeton University Press, Princeton (2005)
[20] de Best, J.J.T.H., Bukkems, B.H.M., Molengraft, M.J.G., Heemels, W.P.M.H., Steinbuch, M.: Robust control of piecewise linear systems: a case study in sheet flow control. Control Engineering Practice 16(8), 991–1003 (2008)
[21] Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, J.: Diagnosis and Fault-Tolerant Control, 2nd edn. Springer, Heidelberg (2006)
[22] Bloemen, H.H.J., van den Boom, T.J.J., Verbruggen, H.B.: Model-based predictive control for Hammerstein-Wiener systems. Int. J. Control 74(5), 482–495 (2001)
[23] Blondel, V.D., Bournez, O., Koiran, P., Tsitsiklis, J.N.: The stability of saturated linear dynamical systems is undecidable. J. Computer and System Sciences 62, 442–462 (2001)
[24] Blondel, V.D., Tsitsiklis, J.N.: A survey of computational complexity results in systems and control. Automatica 36(9), 1249–1274 (2000)
[25] Bodson, M., Groszkiewicz, J.E.: Multivariable adaptive algorithms for reconfigurable flight control. IEEE Trans. Contr. Syst. Technol. 5(2), 217–229 (1997)
[26] Bonivento, C., Isidori, A., Marconi, L., Paoli, A.: Implicit fault-tolerant control: application to induction motors. Automatica 40(3), 355–371 (2004)
[27] Borkar, S.: Designing reliable systems from unreliable components: the challenges of transistor variability and degradation. IEEE Micro 25(6), 10–16 (2005)
[28] Boskovic, J., Mehra, R.: A multiple model-based reconfigurable flight control system design. In: Proc. 37th IEEE Conf. Decision and Control, pp. 4503–4508 (1998)
[29] Boyd, S., Ghaouni, L.E., Feron, E., Balakrishnan, V.: Linear Matrix Inequalities in System and Control Theory. SIAM, Philadelphia (1994)
[30] Boyd, S., Vandenberghe, L.: Convex optimization. Cambridge University Press, Cambridge (2004)
[31] Brasch, F.M., Pearson, J.B.: Pole placement using dynamic compensators. IEEE Trans. Autom. Control AC-15(1), 34–43 (1970)
[32] Bryson, A.E., Luenberger, D.G.: The synthesis of regulator logic using state-variable concepts. Proc. IEEE 58(11), 1803–1811 (1970)

[33] Caglayan, A.K., Allen, S.M., Wehmuller, K.: Evaluation of a second generation reconfiguration strategy for aircraft flight control systems subjected to actuator failure/surface damage. In: Proc. 1988 National Aerospace and Electronics Conference, pp. 520–529 (1988)
[34] Campos-Delgado, D.U., Zhou, K.: Reconfigurable fault-tolerant control using GIMC structure. IEEE Trans. Autom. Control 48(5), 832–839 (2003)
[35] Carmona, V., Freire, E., Ponce, E., Torres, F.: On simplifying and classifying piecewise-linear systems. IEEE Trans. Circuits Syst. I, Fundam. Theory Appl. 49(5), 609–620 (2002)
[36] Caspar, M.: Regelung eines Systems mit örtlich verteilten Parametern mittels virtuellem Aktor. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2009); ATP 0066
[37] Chen, W., Saif, M.: Adaptive actuator fault detection, isolation and accommodation in uncertain systems. Int. J. Control 80(1), 45–63 (2007)
[38] Cho, K.H., Lim, J.T.: Synthesis of fault-tolerant supervisor for automated manufacturing systems: a case study on photolithographic process. IEEE Trans. Robotics and Automation 14(2), 348–351 (1998)
[39] Cho, K.H., Lim, J.T.: Fault-tolerant supervisory control under C, D observability and its application. Int. J. Systems Science 31(12), 1573–1583 (2000)
[40] Cieslak, J., Henry, D., Zolghadri, A.: A methodology for the design of active fault-tolerant control systems. In: Proceedings of the 6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes (Safeprocess), Beijing, China, pp. 865–870 (2006)
[41] Cortes, J.: Discontinuous dynamical systems. IEEE Control Systems Magazine 28(3), 36–73 (2008)
[42] Dangoumau, N., Toguyeni, A.K.A., Craye, E.: Functional and behavioural modelling for dependability in automated production systems. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 216(3), 389–405 (2002)
[43] Darouach, M., Zasadzinski, M., Xu, S.J.: Full-order observers for linear systems with unknown inputs. IEEE Trans. Autom. Control 39(3), 606–609 (1994)
[44] Davies, J., Steffen, T., Dixon, R., Goodall, R.M., Zolotas, A., Pearson, J.: Modelling of high redundancy actuation utilising multiple moving coil actuators. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 3228–3233 (2008)
[45] Demidovich, B.P.: Lectures on Stability Theory, Nauka, Moscow (1967) (in Russian)
[46] Deutsches Institut für Normung (DIN): Din 40041 Zuverlässigkeit - Begriffe (1990)
[47] Diallo, D., Benbouzid, M.E.H., Makouf, A.: A fault-tolerant control architecture for induction motor drives in automotive applications. IEEE Trans. Autom. Control 53(6), 1847–1855 (2004)
[48] Ding, S.X.: Model-based Fault Diagnosis Techniques: Design Schemes, Algorithms, and Tools. Springer, Heidelberg (2008)
[49] Discenzo, F.M., Unsworth, P.J., Loparo, K.A., Marcy, H.O.: Self-diagnosing intelligent motors: a key enabler for nextgeneration manufacturing systems. In: IEE Colloquium on Intelligent and Self-Validating Sensors, Oxford, UK, pp. 3/1–3/4 (1999)
[50] Dong, J., Verhaegen, M., Holweg, E.: Closed-loop subspace predictive control for fault-tolerant MPC design. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 3216–3221 (2008)

[51] Farrell, J., Berger, T., Appleby, B.: Using learning techniques to accommodate unanticipated faults. IEEE Control Systems Magazine 13(3), 40–49 (1993)
[52] Feng, C.C.: Robust fault-tolerant control for systems with extended bounded-sensor-faults. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 2582–2587 (2008)
[53] Feng, G.: An approach to $H_\infty$ controller synthesis of piecewise linear systems. Communications in Information and Systems 2(3), 245–254 (2002)
[54] Ferrari-Trecate, G., Cuzzola, F.A., Mignone, D., Morari, M.: Analysis of discrete-time piecewise affine and hybrid systems. Automatica 38(12), 2139–2146 (2002)
[55] Ferrari-Trecate, G., Mignone, D., Morari, M.: Moving horizon estimation for hybrid systems. IEEE Trans. Automatic Control 47(10), 1663–1676 (2002)
[56] Ferrari-Trecate, G., Muselli, M., Liberati, D., Morari, M.: A clustering technique for the identification of piecewise affine systems. Automatica 39(2), 205–217 (2003)
[57] Fliess, M., Join, C., Sira-Ramirez, H.: Non-linear estimation is easy. Int. J. Modelling Identification and Control 4(1), 12–27 (2008)
[58] Fliess, M., Levine, J., Martin, P., Rouchon, P.: Flatness and defect of nonlinear systems: introductory theory and examples. Int. J. Control 61(6), 1327–1361 (1995)
[59] Fritsch, C., Richter, J.H., Lunze, J., Steffen, T.: Rapid Control Prototyping mit MATLAB/Simulink und SPS. In: Krammer, E. (ed.) Steuerungstechnik aktuell, Oldenbourg Industrieverlag (2007)
[60] Fritsch, C., Richter, J.H., Steffen, T., Lunze, J.: Rapid Control Prototyping mit MATLAB/Simulink und speicher-programmierbaren Steuerungen. ATP Automatisierungstechnische Praxis 48(5), 54–62 (2006)
[61] Fuente, M.J., Mateo, V., Sainz, G.I., Saludes, S.: Adaptive neural-based fault-tolerant control for nonlinear systems. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 2595–2600 (2008)
[62] Fujitsu Microelectronics, Fujitsu Deutschland GmbH, Frankfurter Ring 211, D-80807 München: 32-bit Microcontroller CMOS FR60 MB91460G Series (2008); FME-MB91460G rev 3.0
[63] Ganguli, S., Marcos, A., Balas, G.: Reconfigurable LPV control design for boeing 747/-100/200 longitudinal axis. In: Proc. American Control Conference, Anchorage, USA, vol. 5, pp. 3612–3617 (2002)
[64] Gao, Z., Antsaklis, P.J.: Stability of the pseudo-inverse method for reconfigurable control systems. Int. J. Control 53(3), 717–729 (1991)
[65] Gao, Z., Antsaklis, P.J.: Reconfigurable control system design via perfect model following. Int. J. Control 56(4), 783–798 (1992)
[66] Geisler, J., Witting, K., Trächtler, A., Dellnitz, M.: Multiobjective optimization of control trajectories for the guidance of a rail-bound vehicle. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 4380–4386 (2008)
[67] Gertler, J.: Fault Detection and Diagnosis in Engineering Systems. Marcel Dekker, New York (1998)
[68] Goebel, R., Sanfelice, R.G., Teel, A.R.: Hybrid dynamical systems – robust stability and control for systems that combine continuous-time and discrete-time dynamics. IEEE Control Systems Magazine 29(2), 28–93 (2009)
[69] Goethals, I., Pelckmans, K., Suykens, J.A.K., de Moor, B.: Subspace identification of Hammerstein systems using least squares support vector machines. IEEE Trans. Autom. Control 50(10), 1509–1519 (2005)
[70] Golub, G.H., van Loan, C.F.: Matrix Computations, 3rd edn. The John Hopkins University Press, Baltimore (1996)

[71] Grüne, L., Worthmann, K., Nesic, D.: Continuous-time controller redesign for digital implementation: A trajectory based approach. Automatica 44(1), 225–232 (2008)
[72] Hagenmeyer, V.: Robust nonlinear tracking control based on differential flatness. No. 978 in Reihe 8: Mess- Steuerungs- und Regelungstechnik. VDI-Verlag, Düsseldorf (2003)
[73] Hante, F.M., Leugering, G., Seidman, T.I.: Modeling and analysis of modal switching in networked transport systems. J. Applied Mathematics and Optimization 59(2), 275–292 (2009)
[74] Heemels, W.P.M.H., de Schutter, B., Bemporad, A.: Equivalence of hybrid dynamical models. Automatica 37(7), 1085–1091 (2001)
[75] Heemels, W.P.M.H., Weiland, S., Juloski, A.L.: Input-to-state stability of discontinuous dynamical systems with an observer-based control application. In: Bemporad, A., Bicchi, A., Buttazzo, G. (eds.) HSCC 2007. LNCS, vol. 4416, pp. 259–272. Springer, Heidelberg (2007)
[76] Hespanha, J.P., Naghshtabrizi, P., Xu, Y.: A survey of recent results in networked control systems. Proc. of IEEE Special Issue on Technology of Networked Control Systems 95(1), 138–162 (2007)
[77] Hilsch, M., Lunze, J., Nitsche, R.: Fault-tolerant internal model control with application to a diesel engine. In: Proc. 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Barcelona, Spain, pp. 1091–1096 (2009); ThC5.3
[78] Hirche, S., Lunze, J.: Digital vernetzte Regelungssysteme. Automatisierungstechnik 56(1), 1–3 (2008)
[79] Hosford, J.E.: Measures of dependability. Operations Research 8(1), 53–64 (1960)
[80] Hou, M., Müller, P.C.: Design of observers for linear systems with unknown inputs. IEEE Trans. Autom. Control 37(6), 871–875 (1992)
[81] Incropera, F.P., DeWitt, D.P.: Fundamentals of Heat and Mass Transfer, 6th edn. Wiley & Sons, Chichester (2006)
[82] Ingimundarson, A., Peña, R.S.S.: Using the unfalsified control concept to achieve fault tolerance. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1236–1242 (2008)
[83] Isermann, R.: Fault-Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance. Springer, Heidelberg (2006)
[84] Isidori, A., Byrnes, C.I.: Output regulation of nonlinear systems. IEEE Trans. Autom. Control 35(2), 131–140 (1990)
[85] Jakubowski, T.: Erweiterung der Sicherheitsüberwachung für die verfahrenstechnische Pilotanlage. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2005); ATP 0033
[86] Jiang, B., Staroswiecki, M., Cocquempot, V.: Fault accommodation for nonlinear dynamic systems. IEEE Trans. Autom. Control 51(9), 1578–1583 (2006)
[87] Jiang, J.: Fault-tolerant control systems – an introductory overview. Acta Automatica Sinica 31(1), 161–174 (2005)
[88] Jiang, J., Zhang, Y.: Fault diagnosis and reconfigurable control of a pressurizer in a nuclear power plant. In: Proc. 5th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, Washington D.C., USA, pp. 1095–1100 (2003)
[89] Jiang, Z.P., Teel, A.R., Praly, L.: Small-gain theorem for ISS systems and applications. Mathematics of Control, Signals, and Systems 7, 95–120 (1994)
[90] Johansson, M.: Piecewise Linear Control Systems. LNCIS, vol. 284. Springer, Heidelberg (2003)

[91] Johansson, M., Rantzer, A.: Computation of piecewise quadratic Lyapunov functions for hybrid systems. IEEE Trans. Autom. Control 43(4), 555–559 (1998)
[92] Juloski, A.L., Heemels, W.P.M.H., Ferrari-Trecate, G., Vidal, R., Paoletti, S., Niessen, J.H.G.: Comparison of four procedures for identification of hybrid systems. In: Morari, M., Thiele, L. (eds.) HSCC 2005. LNCS, vol. 3414, pp. 354–369. Springer, Heidelberg (2005)
[93] Juloski, A.L., Heemels, W.P.M.H., Weiland, S.: Observer design for a class of piecewise linear systems. Int. J. Robust Nonlinear Control 17(15), 1387–1404 (2007)
[94] Juloski, A.L., Heemels, W.P.M.H., Weiland, S.: Output feedback control for a class of piecewise linear systems. In: Proc. American Control Conference, New York, U.S.A, pp. 1383–1388 (2007)
[95] Juloski, A.L., Weiland, S., Heemels, W.P.M.H.: A bayesian approach to identification of hybrid systems. In: Proc. 43th IEEE Conference on Decision and Control (CDC), pp. 13–19 (2004)
[96] Kalman, R.: Mathematical description of linear dynamical systems. SIAM J. Control, 152–192 (1963)
[97] Khalil, H.K.: Nonlinear Systems, 3rd edn. Prentice-Hall, New Jersey (2002)
[98] Krokavec, D., Filasová, A.: Performance of reconfiguration structures based on the constrained control. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1243–1248 (2008)
[99] Lazar, M., de la Pena, D.M., Heemels, W., Alamo, T.: On the stability of min-max nonlinear model predictive control. Systems and Control Letters 57(1), 39–48 (2008)
[100] LeBel, S., Rodrigues, L.: PWL and PWA $H_\infty$ controller synthesis for uncertain PWA slab systems: LMI approach. Int. J. Control 82(3), 482–492 (2009)
[101] Lehmann, D.: Modellbildung und Identifikation eines Batchreaktors. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2006); ATP 0039
[102] Leuer, C.: Diagnose und Rekonfiguration von Sensorfehlern mittels virtuellem Sensor. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2006); ATP 0038
[103] Li, Z., Zolotas, A.C., Jaimoukha, I.M., Grigoriadis, K.M.: Output selection with fault tolerance via dynamic controller design. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 2570–2575 (2008)
[104] Liberzon, D.: Switching in Systems and Control. In: Systems & Control: Foundations and Applications (2003)
[105] Lin, H., Antsaklis, P.J.: Stability and stabilizability of switched linear systems: A survey of recent results. IEEE Trans. Autom. Control 54(2), 308–322 (2009)
[106] Lindegaard, K.P., Fossen, T.I.: Fuel-efficient rudder and propeller control allocation for marine craft: experiments with a model ship. IEEE Trans. Contr. Syst. Technol. 11(6), 850–862 (2003)
[107] Lions, J.L.: Pointwise control for distributed systems. In: Banks, H.T. (ed.) Control and Estimation in Distributed Parameter Systems. Frontiers in Applied Mathematics, vol. 11, pp. 1–39. SIAM, Philadelphia (1992)
[108] Løvaas, C., Seron, M.M., Goodwin, G.C.: Robust output-feedback MPC with soft state constraints. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 13, 157–162 (2008)

[109] Löfberg, J.: Yalmip : A toolbox for modeling and optimization in MATLAB. In: Proceedings of the CACSD Conference, Taipei, Taiwan (2004), http://control.ee.ethz.ch/~joloef/yalmip.php
[110] Luenberger, D.G.: Observing the state of a linear system. IEEE Trans. Military Electronics (8), 74–80 (1964)
[111] Luenberger, D.G.: Observers for multivariable systems. IEEE Trans. Autom. Control AC-11(2), 190–197 (1966)
[112] Luenberger, D.G.: An introduction to observers. IEEE Trans. Autom. Control 16(6), 596–602 (1971)
[113] Luenberger, D.G.: A double look at duality. IEEE Trans. Autom. Control 37(10), 1474–1482 (1992)
[114] Luenberger, D.G.: Information Science. Princeton University Press, New York (2006)
[115] Lunze, J.: Control reconfiguration after actuator failures: the generalised virtual actuator. In: Proc. 6th IFAC Sympos. on Fault Detection, Supervision and Safety for Technical Processes, Beijing, China, pp. 1309–1314 (2006)
[116] Lunze, J.: Fault diagnosis of discretely controlled continuous systems by means of discrete-event models. Discrete Event Dynamic Systems: Theory and Applications 18(2), 181–210 (2008)
[117] Lunze, J.: Regelungstechnik 2, 5th edn. Springer, Heidelberg (2008)
[118] Lunze, J., Lamnabhi-Lagarrigue, F. (eds.): The HYCON handbook of hybrid systems: control, theory, applications. Cambridge University Press, Cambridge (2009) (to appear)
[119] Lunze, J., Richter, J.H.: Design approach to the generalised virtual actuator for control reconfiguration. Automatisierungstechnik 54(7), 353–361 (2006)
[120] Lunze, J., Richter, J.H.: Reconfigurable fault-tolerant control: a tutorial introduction. European J. Control 14(5), 359–386 (2008)
[121] Lunze, J., Steffen, T.: Reconfigurable control of a quantised system. In: Proc 4th Symposium on Fault Detection, Supervision and Safety for Technical Processes, Budapest, Hungary, pp. 822–827 (2000)
[122] Lunze, J., Steffen, T.: Hybrid reconfigurable control. In: Engell, S., Frehse, G., Schnieder, E. (eds.) Modelling, Analysis and Design of Hybrid Systems. LNCIS, vol. 279, pp. 267–284. Springer, Heidelberg (2002)
[123] Lunze, J., Steffen, T.: Control reconfiguration by means of a virtual actuator. In: Proc. IFAC Sympos. on Fault Detection, Supervision and Safety for Technical Processes. Washington, U.S.A. (2003)
[124] Lunze, J., Steffen, T.: Rekonfiguration linearer Systeme bei Aktor- und Sensorfehlern. Automatisierungstechnik 51, 60–68 (2003)
[125] Lunze, J., Steffen, T.: Control reconfiguration after actuator failures using disturbance decoupling methods. IEEE Trans. Autom. Control 51(10), 1590–1601 (2006)
[126] Luong, M., Maquin, D., Huynh, C.T., Ragot, J.: Observability, redundancy, reliability and integrated design of measurement systems. In: Proc. of 2nd IFAC Sympos. Intelligent Components and Instruments for Control Applications, Budapest, Hungary (1994)
[127] Maciejowski, J.M., Jones, C.N.: MPC fault-tolerant flight control case study: Flight 1862. In: Proc. 5th IFAC Sympos. Fault Detection, Supervision and Safety for Techn. Processes, M1-C1, Washington D.C., USA, pp. 121–126 (2003)
[128] Mahmoud, M., Jiang, J., Zhang, Y.: Active Fault Tolerant Control Systems. Springer, Heidelberg (2003)

[129] Martinez, J.J., Seron, M.M., de Dona, J.: Fault-tolerant switching scheme with multiple sensor-controller pairs. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1212–1217 (2008)
[130] Mendonça, L.F., Sousa, J.M.C., Sá da Costa, J.M.G.: Fault tolerant control using fuzzy MPC. In: Proc. 6th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes, Beijing, China, pp. 1501–1506 (2006)
[131] Meyer, D.: Virtueller Aktor für Hammerstein-Systeme: Sollwertfolge und Übergangsverhalten. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2008); ATP 0056 (2008)
[132] Morari, A.B.M.: Control of systems integrating logic, dynamics, and constraints. Automatica 35(3), 407–427 (1999)
[133] Morari, M., Baotic, M., Borrelli, F.: Hybrid systems modeling and control. European Journal of Control 9(2), 177–189 (2003)
[134] Murdoch, P.: Dual-observer design procedure. Electronics Letters 9(12), 270–271 (1973)
[135] Münz, E., Krebs, V.: Identification of hybrid systems using a priori knowledge. In: Proc. 15th Triennial World Congress (2002)
[136] Münz, E., Krebs, V.: Continuous optimization approaches to the identification of piecewise affine systems. In: Proc. 17th IFAC World Congress, Seoul, Korea (2008)
[137] Nakada, H., Takaba, K., Katayama, T.: Identification of piecewise affine systems based on statistical clustering technique. Automatica 41(5), 905–913 (2005)
[138] Nayebpanah, N., Rodrigues, L., Zhang, Y.: Fault identification and reconfigurable control for bimodal piecewise affine systems. In: Proc. European Control Conference, Budapest, Hungary, pp. 2694–2699 (2009)
[139] Necoara, I., Schutter, B.D., Heemels, W., Weiland, S., Lazar, M., van den Boom, T.: Control of PWA systems using a stable receding horizon method. In: Proc. 16th IFAC World Congress. Prague, Czech Republic (2005); Paper Tu-E21-TO/2
[140] Neidig, J.: An Automata Theoretic Approach to Modular Diagnosis of Discrete-Event Systems. Jörg Neidig (2007)
[141] Niemann, H., Stoustrup, J.: An architecture for fault tolerant controllers. Int. J. Control 78(14), 1091–1110 (2005)
[142] Nijmeijer, H., van der Schaft, A.: The disturbance decoupling problem for nonlinear control systems. IEEE Trans. Autom. Control AC 28(5), 621–623 (1983)
[143] Nijmeijer, H., van der Schaft, A.: Nonlinear Dynamical Control Systems. Springer, Heidelberg (1996)
[144] Noura, H., Sauter, D., Hamelin, F., Theilliol, D.: Fault-tolerant control in dynamic systems: Application to a winding machine. IEEE Control Systems Magazine 20(1), 33–49 (2000)
[145] Noura, H., Theilliol, D., Ponsart, J.C., Chamseddine, A.: Fault-tolerant Control Systems. Advances in Industrial Control. Springer, Heidelberg (2009)
[146] Ocampo-Martinez, C., de Dona, J., Seron, M.M.: Actuator fault-tolerant control based on invariant set separation. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 7276–7281 (2008)
[147] Olaru, S., de Dona, J., Seron, M.M.: Positive invariant sets for fault tolerant multisensor control schemes. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1224–1229 (2008)
[148] O'Reilly, J.: Observers for linear systems. Academic Press, London (1983)

[149] Ortmann, C.: Anwendung des virtuellen Aktors auf einen nichtlinearen thermofluiden Prozess. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2007); ATP 0046
[150] Palanthandalam, H.J., Bernstein, D.S., Ridley, A.J.: Space weather forcasting. IEEE Control Systems Magazine 27(5), 109–123 (2007)
[151] Paoli, A., Lafortune, S.: Safe diagnosability for fault-tolerant supervision of discrete-event systems. Automatica 41(8), 1335–1347 (2005)
[152] Paoli, A., Sartini, M., Lafortune, S.: A fault tolerant architecture for supervisory control of discrete-event systems. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 6542–6547 (2008)
[153] Parisini, T., Sacone, S.: Fault diagnosis and controller re-configuration: an hybrid approach. In: Proc. 1998 IEEE ISIC/CIRA/ISAS Joint Conf., Gaithersburg, USA, pp. 163–168 (1998)
[154] Park, H.C., Sung, S.W., Lee, J.: Modeling of Hammerstein-Wiener processes with special input test signals. Ind. Eng. Chem. Res. 45(3), 1029–1038 (2006)
[155] Park, S.J., Cho, K.H.: Supervisory control for fault-tolerant scheduling of real-time multiprocessor systems with aperiodic tasks. Int. J. Control 82(2), 217–227 (2009)
[156] Pavlov, A., Pogromsky, A., van de Wouw, N., Nijmeijer, H.: On convergence properties of piecewise affine systems. Int. J. Control 80(8), 1233–1247 (2007)
[157] Pavlov, A., Pogromsky, A., van der Wouw, N., Nijmeijer, H., Rooda, K.: Convergent piecewise affine systems: analysis and design. Part II: discontinuous case. In: Proc. 44th IEEE Conf. Decision and Control and European Control Conf., Seville, Spain, pp. 5397–5402 (2005)
[158] Pavlov, A., van der Wouw, N., Nijmeijer, H.: Convergent piecewise affine systems: analysis and design. Part I: continuous case. In: Proc. 44th IEEE Conf. Decision and Control and European Control Conf., Seville, Spain, pp. 5391–5396 (2005)
[159] Pavlov, A., van de Wouw, N., Nijmeijer, H.: Uniform output regulation of nonlinear systems. Systems & Control: Foundations and Applications. Birkhäuser (2006)
[160] Pettersen, K.Y., Mazenc, F., Nijmeijer, H.: Global uniform asymptotic stabilization of an underactuated surface vessel: experimental results. IEEE Trans. Contr. Syst. Technol., 891–903 (2004)
[161] Pham, H.: Handbook of Reliability Engineering. Springer, Heidelberg (2003)
[162] Phillips, J.: Diagnosis and reconfiguration experiments on VERA. Project, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2006); Report 2006 17
[163] Planchon, P.: Guaranteed diagnosis of uncertain linear systems using state-set observation. Logos-Verlag, Berlin (2007)
[164] Prasad, D., McDermid, J., Wand, I.: Dependability terminology: Similarities and differences. IEEE AES Systems Magazine 11(1), 14–21 (1996)
[165] Rantzer, A., Johansson, M.: Piecewise linear optimal control. IEEE Trans. Autom. Control 45(4), 629–637 (2000)
[166] Reissenweber, B.: Feldbussysteme zur industriellen Kommunikation, 2nd edn. Oldenbourg Wissenschaftsverlag (2001)
[167] Richter, J.H.: Control of distributed-parameter systems by means of virtual actuators. Forschungsbericht, Institute of Automation and Computer Control, Ruhr-Universität, Bochum, Germany (2009)

[168] Richter, J.H.: Control reconfiguration toolbox: user's and developer's manual. Forschungsbericht, Institute of Automation and Computer Control, Ruhr-Universität, Bochum, Germany (2009)
[169] Richter, J.H., Heemels, W.P.M.H., van de Wouw, N., Lunze, J.: Reconfigurable control of PWA systems with actuator and sensor faults: stability. In: Proceedings of the 47th Conf. Decision and Control (CDC), Cancun, Mexico, pp. 1060–1065 (2008); TuB13.4
[170] Richter, J.H., Heemels, W.P.M.H., van de Wouw, N., Lunze, J.: Reconfigurable control of PWA systems with actuator and sensor faults: stability and tracking. Automatica (accepted, 2010)
[171] Richter, J.H., Lunze, J.: Markov-parameter-based control reconfiguration by matching the I/O-behaviour of the plant. In: Proc. European Contr. Conf., ECC 2007, Kos, Greece, pp. 2942–2949 (2007); WeA11.3
[172] Richter, J.H., Lunze, J.: Reconfigurable control of Hammerstein systems after actuator faults. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 3210–3215 (2008)
[173] Richter, J.H., Lunze, J.: $H_\infty$-based virtual actuator synthesis. In: Proc. 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes, pp. 1587–1592. IFAC, Barcelona (2009); FrC4.2
[174] Richter, J.H., Lunze, J.: Reconfigurable control of Hammerstein systems after actuator failures: stability, tracking, and performance. Int. J. Control 83(8), 1612–1630 (2010)
[175] Richter, J.H., Lunze, J., Schlage, T.: Control reconfiguration after actuator failures by Markov parameter matching. Int. J. Control 81(9), 1382–1398 (2008)
[176] Richter, J.H., Lunze, J., Steffen, T.: The relation between the virtual actuator and the dual observer. European J. Control 16(5) (2010)
[177] Richter, J.H., Schlage, T., Lunze, J.: Control reconfiguration of a thermofluid process by means of a virtual actuator. IET Control Theory Appl. 1(6), 1606–1620 (2007)
[178] Richter, J.H., Schlage, T., Lunze, J.: Rekonfiguration eines thermofluiden Prozesses mittels virtuellem Aktor. Automatisierungstechnik 55(4), 157–169 (2007)
[179] Richter, J.H., Weiland, S., Heemels, W.P.M.H., Lunze, J.: Decoupling-based reconfigurable control of linear systems after actuator faults. In: Proc. 10th European Control Conference. Budapest, Hungary (2009); TuB6.2
[180] Richter, J.H., Weiland, S., Heemels, W.P.M.H., Lunze, J.: Reconfigurable control of linear systems with actuator and sensor faults: geometric solutions (2009) (in preparation)
[181] Rodrigues, M., Theilliol, D., Sauter, D.: Fault tolerant control design for switched systems. In: Proc. 2nd IFAC Conf. on Analysis and Design of Hybrid Systems, pp. 223–228. Alghero, Italy (2006)
[182] Roll, J., Bemporad, A., Ljung, L.: Identification of piecewise affine systems via mixed-integer programming. Automatica 40(1), 37–50 (2004)
[183] Rosenbrock, H.H.: Distinctive problems of process control. Chemical Engineering Progress 58(9), 43–50 (1962)
[184] Rosich, A., Puig, V., Quevedo, J.: Fault-tolerant constrained MPC of PEM fuel cells. In: Proc. 46th IEEE Conf. Decision and Control, pp. 2657–2662 (2007)
[185] Saberi, A., Lin, Z., Teel, A.R.: Control of linear systems with saturating actuators. IEEE Trans. Autom. Control 41(3), 368–378 (1996)
[186] Sastry, S.: Nonlinear systems – Analysis, stability and control. In: Interdisciplinary Applied Mathematics, vol. 10. Springer, Heidelberg (1999)
[187] Scherer, C., Weiland, S.: Linear Matrix Inequalities in Control. Delft Center for Systems and Control (2005)

[188] Schlage, T.: Rekonfiguration einer Prozessregelung mittels virtuellem Aktor. Master thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2006); ATP 0035
[189] Schröder, J.: Modelling, State Observation and Diagnosis of Quantised Systems. Springer, Heidelberg (2003)
[190] Schulz, M.: Modellbildung und Identifikation des Wärmetauschers X1. Bachelor thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2006); ATP 0036
[191] Schumacher, J.M.: Compensator synthesis using (C,A,B)-pairs. IEEE Trans. Autom. Control AC-25(6), 1133–1138 (1980)
[192] de Schutter, B., van den Boom, T.: MPC for continuous piecewise-affine systems. Systems and Control Letters 52(3-4), 179–192 (2004)
[193] Sebek, M., Kucera, V.: Polynomial approach to quadratic tracking in discrete linear systems. IEEE Trans. Autom. Control AC-27(6), 1248–1250 (1982)
[194] Selmic, R.R., Polycarpou, M.M., Parisini, T.: Actuator fault detection in nonlinear uncertain systems using neural on-line approximation models. European J. Control 15(1), 29–44 (2009)
[195] Seron, M.M., de Dona, J.: Fault tolerant control using virtual actuators and invariant-set based fault detection and identification. In: 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference, Shanghai, China (2009) (to appear)
[196] Seron, M.M., Romero, M., de Dona, J.: Sensor fault tolerant control of induction motors. In: Proc. 17th IFAC World Congress, pp. 1230–1235. Seoul, Korea (2008)
[197] Sontag, E.D.: Nonlinear regulation: The piecewise linear approach. IEEE Trans. Autom. Control 26(2), 346–358 (1981)
[198] Sontag, E.D.: Smooth stabilisation implies coprime factorization. IEEE Trans. Autom. Control 34(4), 435–443 (1989)
[199] Sontag, E.D.: On the input-to-state stability property. European J. Control 1, 24–36 (1995)
[200] Sontag, E.D.: The ISS philosophy as a unifying framework for stability-like behavior. In: Nonlinear Control in the Year 2000. LNCIS, vol. 259, pp. 443–467. Springer, Heidelberg (2001)
[201] Staroswiecki, M., Yang, H., Jiang, B.: Progressive accommodation of aircraft actuator faults. In: Proc. 6th IFAC Sympos. Fault Detection, Supervision and Safety for Techn. Processes, Beijing, China, pp. 877–882 (2006)
[202] Steffen, T.: Control Reconfiguration of Dynamical Systems: Linear Approaches and Structural Tests. LNCIS, vol. 320. Springer, Heidelberg (2005)
[203] Steffen, T., Davies, J., Dixon, R., Goodall, R.M., Pearson, J., Zolotas, A.: Failure modes and probabilities of a high redundancy actuator. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 3234–3239 (2008)
[204] Stengel, R.F.: Intelligent failure-tolerant control. IEEE Control Systems Magazine 11(4), 14–23 (1991)
[205] Stengel, R.F.: Flight Dynamics. Princeton University Press, USA (2004)
[206] Stoustrup, J., Blondel, V.D.: Fault tolerant control: A simultaneous stabilization result. IEEE Trans. Autom. Control 49(2), 305–310 (2004)
[207] Sturm, J.: Using Sedumi 1.02, a Matlab toolbox for optimization over symmetric cones. Optimization Methods and Software 11, 625–653 (1999)
[208] Supavatanakul, P.: Modelling and Diagnosis of Timed Discrete-Event Systems. Shaker-Verlag, Aachan (2004)

[209] Theilliol, D., Ponsart, J.C., Rodrigues, M., Aberkane, S., Yamé, J.: Design of sensor fault diagnosis method for nonlinear systems described by linear polynomial matrices formulation: application to a winding machine. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1890–1895 (2008)
[210] Tran, T.H., Stursberg, O., Engell, S.: Reconfiguration of discretely controlled hybrid systems for changing specifications. In: Proc. 1st IFAC Workshop on Dependable Control of Discrete Systems, pp. 247–252. ENS-Cachan, Paris, France (2007)
[211] Trentelman, H., Stoorvogel, A.A., Hautus, M.: Control Theory for Linear Systems. Perspectives in Neural Computing. Springer, Heidelberg (2001)
[212] Tsai, J.S.H., Lee, Y.Y., Cofie, P., Chen, L.S.S.X.M.: Active fault tolerant control using state-space self-tuning control approach. Int. J. Systems Science 11(15), 785–797 (2006)
[213] Tsuda, K., Mignone, D., Ferrari-Trecate, G., Morari, M.: Reconfiguration strategies for hybrid systems. In: Proc. 2001 American Control Conference, Arlington, VA, USA, vol. 2, pp. 868–873 (2001)
[214] Vatanski, N., Georges, J.P., Aubrun, C., Jämsä-Jounela, S.L.: Control reconfiguration in networked control system. In: Proc. 6th IFAC Sympos. Fault Detection, Supervision and Safety for Techn. Processes, PRC, Beijing (2006)
[215] Vidal, R.: Identification of PWARX hybrid models with unknown and possibly different orders. In: Proc. 43th American Control Conference (ACC), pp. 547–552 (2004)
[216] von Neumann, J.: Probabilistic logic and the synthesis of reliable organisms from unreliable components. In: Shannon, C., McCarthy, J. (eds.) Automata Studies, Annals of Mathematical Studies, vol. 34, pp. 43–98 (1956)
[217] Wang, F.Y., Liu, D.: Networked Control Systems. Springer, London (2008)
[218] Wang, Y.: Erprobung und Vergleich von Eingangs-Beobachtern für die Aktordiagnose. Master thesis, Lehrstuhl für Automatisierungstechnik und Prozessinformatik, Fakultät für Elektrotechnik und Informationstechnik, Ruhr-Universität Bochum (2007); ATP 0044
[219] Weiland, S., Willems, J.: Almost disturbance decoupling with internal stability. IEEE Trans. Autom. Control 34(3), 277–285 (1989)
[220] Wen, C., Wang, S., Jin, X., Ma, X.: Identification of dynamic systems using piecewise-affine basis function models. Automatica 43(10), 1824–1831 (2007)
[221] Weng, Z., Patton, R., Cui, P.: Integrated design of robust controller and fault estimator for linear parameter varying systems. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 4535–4539 (2008)
[222] Willems, J.: Almost invariant subspaces: an approach to high gain feedback design – Part I: Almost controlled invariant subspaces. IEEE Trans. Autom. Control AC-26(1), 235–252 (1981)
[223] Willems, J.: Almost invariant subspaces: an approach to high gain feedback design – Part II: almost conditionally invariant subspaces. IEEE Trans. Autom. Control AC-27(5), 1071–1085 (1982)
[224] Willems, J.C.: The behavioural approach to open and interconnected systems – modelling by tearing, zooming, and linking. IEEE Control Systems Magazine 27(6), 46–99 (2007)
[225] Wolff, F., Krutina, P., Krebs, V.: Robust consistency-based diagnosis of nonlinear systems by set observation. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 10, 124–129 (2008)
[226] Wonham, W.M.: On pole assignment in multi-input controllable linear systems. IEEE Trans. Autom. Control AC-12, 660–665 (1967)

[227] Wonham, W.M.: Linear Multivariable Control: a Geometric Approach. In: Applications of Mathematics, 2nd edn., Springer, Heidelberg (1979)
[228] van de Wouw, N., Pavlov, A.: Tracking and synchronisation for a class of PWA systems. Automatica 44(11), 2909–2915 (2008)
[229] Wu, N.E., Thavamani, S., Zhang, Y., Blanke, M.: Sensor fault masking of a ship propulsion system. Control Engineering Practice 14(11), 1337–1345 (2006)
[230] Xiushan, F., Chengde, H.: A fault-tolerant routing scheme in dynamic networks. J. Comput. Sci. & Technol. 16(4), 371–380 (2001)
[231] Xu, J., Xie, L.: Homogeneous polynomial Lyapunov functions for piecewise affine systems. In: Proc. American Control Conf., Portland, OR, U.S.A, pp. 581–586 (2005)
[232] Yamé, J., Sauter, D., Aubrun, C.: Design of fault-tolerant controllers for guaranteed $H_2$-performance over digital networks with time-varying communication delays. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1273–1278 (2008)
[233] Yang, H., Cocquempot, V., Jiang, B.: Fault tolerance analysis for switched systems via global passivity. Int. J. Control 82(1), 117–129 (2009)
[234] Yang, H., Jiang, B., Cocquempot, V.: Fault accommodation for discrete event systems using Petri net with application to traffic light control. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1218–1223 (2008)
[235] Yang, H., Jiang, B., Cocquempot, V.: A fault tolerant control framework for periodic switched non-linear systems. Int. J. Control 82(1), 117–129 (2009)
[236] Yang, H., Jiang, B., Cocquempot, V., Staroswiecki, M.: Adaptive fault tolerant strategy for hybrid systems with faults independently affecting on outputs. In: Proc. 6th IFAC Sympos. Fault Detection, Supervision and Safety for Techn. Processes, Beijing, China, pp. 1021–1026 (2006)
[237] Yang, H., Jiang, B., Staroswiecki, M.: Observer-based fault-tolerant control for a class of switched nonlinear systems. IET Control Theory Appl. 1(5), 1523–1532 (2007)
[238] Yang, H., Mao, Z.H., Jiang, B.: Model-based fault tolerant control for hybrid dynamic systems with sensor faults. Acta Automatica Sinica 32(5), 680–685 (2006)
[239] Yang, K.C.H., Yuh, J., Choi, S.K.: Fault-tolerant system design of an autonomous underwater vehicle - ODIN: an experimental study. Operations Research 8(1), 53–64 (1960)
[240] Yang, S.S., Chen, J., Mohamed, H.A.F., Moghavvemi, M.: Sensor fault tolerant controller for a double inverted pendulum system. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 2588–2594 (2008)
[241] Yang, Z.: A hybrid system approach towards redundant fault-tolerant control systems. In: Proc. 39th IEEE Conf. Decision and Control, Sydney, Australia, pp. 987–992 (2000)
[242] Yang, Z., Blanke, M., Verhaegen, M.: Robust control mixer method for reconfigurable control design using model matching. IET Control Theory Appl. 1(1), 349–357 (2007)
[243] Yang, Z., Stoustrup, J.: Robust reconfigurable control for parametric and additive faults with FDI uncertainties. In: Proc. 39th IEEE Conf. Decision and Control, Sydney, Australia, pp. 4132–4137 (2000)
[244] Yepremyana, L., Falk, J.E.: Delaunay partitions in $\mathbb{R}^n$ applied to non-convex programs and vertex/facet enumeration problems. Computers & Operations Research 32(4), 793–812 (2005)
[245] Youla, D.C., Bongiorno Jr., J.J.: A feedback theory of two-degree-of-freedom optimal Wiener-Hopf design. IEEE Trans. Autom. Control AC-30(7), 652–664 (1985)

[246] Youla, D.C., Jabr, H.A., Bongiorno Jr., J.J.: Modern Wiener-Hopf design of optimal controllers – part II: the multivariable case. IEEE Trans. Autom. Control AC-21(3), 319–337 (1976)
[247] Zhang, L., Andreeski, C., Dimirovski, G.M., Jing, Y.W.: Reliable adaptive control for switched fuzzy systems. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 7636–7641 (2008)
[248] Zhang, X., Polycarpou, M.M., Parisini, T.: A robust detection and isolation scheme for abrupt and incipient faults in nonlinear systems. IEEE Trans. Autom. Control 47(4), 576–593 (2002)
[249] Zhang, Y., Jiang, J.: Bibliographical review on reconfigurable fault-tolerant control systems. In: Proc. 5th IFCAC Sympos. Fault Detection, Supervision and Safety for Techn. Processes, M2-C1, Washington D.C., USA, pp. 265–276 (2003)
[250] Zhang, Y., Jiang, J.: Issues on integration of fault diagnosis and reconfigurable control in active fault-tolerant control systems. In: Proc. 6th IFAC Symposium on Fault Detection Supervision and Safety for Technical Processes, Beijing, China, pp. 1513–1524 (2006)
[251] Zhang, Y., Qin, S.J.: Adaptive actuator/component fault compensation for nonlinear systems. AIChe Journal 54(9), 2404–2412 (2008)
[252] Zhang, Y., Wu, N.E., Jiang, B.: Fault detection and isolation applied to a ship propulsion benchmark. In: Proc. 17th IFAC World Congress, Seoul, Korea, pp. 1908–1913 (2008)
[253] Ziegler, G.M.: Lectures on Polytopes, revised 1st edn. Springer, Heidelberg (1998)

# Part V
# Appendices

# Appendix A
# Acronyms and Symbols

**Acronyms**

| | |
|---|---|
| ECU | Electronic control unit |
| FDI | Fault detection and isolation |
| FTC | Fault-tolerant control |
| GAS | Global asymptotic stability |
| $\delta$GAS | Incremental global asymptotic stability |
| GES | Global exponential stability |
| IOS | Input-to-output stability |
| ISpS | Input-to-state practical stability |
| ISS | Input-to-state stability |
| LMI | Linear matrix inequality |
| ODE | Ordinary differential equation |
| PLC | Programmable logic controller |
| PWA | Piecewise affine |
| PWL | Piecewise linear |
| PWQ | Piecewise quadratic |

**Fields, classes and sets**

| | |
|---|---|
| $\mathbb{C}$ | Field of complex numbers |
| $\mathbb{R}$ | Field of real numbers |
| $\mathcal{C}^\infty$ | Class of smooth functions |
| $\mathcal{K}$ | Class of nondecreasing functions |
| $\mathcal{K}_\infty$ | Class of nondecreasing unbounded functions |
| $\mathcal{KL}$ | Class of functions unbounded and nondecreasing in the first argument and asymptotically decreasing in the second argument |
| $\mathcal{L}_1^{loc}$ | Class of locally integrable functions |
| $\mathcal{L}_2$ | Class of square integrable functions |
| $\mathcal{L}_\infty$ | Class of essentially bounded functions |
| $\Lambda$ | Polyhedron |
| $\mathcal{D}$ | Delaunay partition |

| | |
|---|---|
| $\mathcal{P}$ | Polytope |
| $\mathcal{S}$ | Subspace of state space |
| $\mathcal{V}$ | Subspace of state space |
| $\mathcal{U}$ | Set of feasible inputs to a system |
| $\mathcal{X}$ | Set of vectors |
| $\mathcal{Z}$ | Set of feasible output equilibria of a system |

**Scalars and scalar-valued functions**

| | |
|---|---|
| $k$ | Dimension of disturbance signal space |
| $m$ | Dimension of input space |
| $n$ | Dimension of state space |
| $p$ | Dimension of controlled output space |
| $q$ | Dimension of measured output space |
| $r$ | Cardinality of state-space partition |
| $w$ | Dimension of reconfiguration block state space |
| $s$ | Complex frequency |
| $\alpha$ | Scalar real-valued function in class $\mathcal{K}$ or class $\mathcal{K}_\infty$ |
| $\beta$ | Scalar real-valued function in class $\mathcal{KL}$ |
| $\gamma$ | Norm of a transfer function |
| $\vartheta$ | Temperature of a fluid |
| $\kappa$ | Number of independently controllable outputs of a dynamical system |
| $\lambda$ | Scalar weight between performance and input effort |
| $\mu$ | Singular value of a matrix |
| $\nu$ | Electrical conductivity of a fluid |
| $\rho$ | Unit step function |
| $\sigma$ | Set of eigenvalues of a matrix |
| $\psi$ | Heading of a vehicle |

**Vectors and vector-valued functions**

| | |
|---|---|
| $\boldsymbol{a}$ | Affine term |
| $\boldsymbol{b}$ | Column of input matrix |
| $\boldsymbol{c}$ | Row of output matrix |
| $\boldsymbol{d}$ | Disturbance input of a dynamical system |
| $\boldsymbol{e}$ | Observation error |
| $\boldsymbol{f}$ | Nonlinear input function |
| $\boldsymbol{g}$ | Nonlinear vector field |
| $\boldsymbol{h}$ | Nonlinear output function |
| $\boldsymbol{k}$ | Vector of sector bounds of nonlinear characteristic; offset vector in polyhedron representation |
| $\boldsymbol{r}$ | Reference input to a control loop |
| $\boldsymbol{s}$ | State of dual observer |
| $\boldsymbol{u}$ | Control input of a dynamical system |
| $\boldsymbol{x}$ | State of a dynamical system |
| $\boldsymbol{x}_\Delta$ | Difference state in reconfiguration block |

| | |
|---|---|
| $\boldsymbol{y}$ | Measured output of a dynamical system |
| $\boldsymbol{y}_{\Delta}$ | Measured output difference between nominal and faulty system |
| $\boldsymbol{z}$ | Controlled output of a dynamical system |
| $\boldsymbol{z}_{\Delta}$ | Controlled output difference between nominal and faulty system |
| $\boldsymbol{\xi}, \boldsymbol{\eta}$ | Auxiliary signals |
| $\boldsymbol{\varphi}$ | Decomposed sector-bounded function |

**Matrices**

| | |
|---|---|
| $\boldsymbol{A}$ | System matrix of a dynamical system |
| $\boldsymbol{B}$ | Input matrix of a dynamical system |
| $\boldsymbol{B_d}$ | Disturbance matrix of a dynamical system |
| $\boldsymbol{C}$ | Measured output matrix of a dynamical system |
| $\boldsymbol{C_z}$ | Controlled output matrix of a dynamical system |
| $\boldsymbol{D}$ | Feedthrough matrix of a dynamical system |
| $\boldsymbol{H}$ | Matrix of normal directions defining a polyhedron |
| $\boldsymbol{I_n}$ | Identity matrix of dimension $n$ |
| $\boldsymbol{K}$ | Static controller feedback matrix |
| $\boldsymbol{L}$ | Virtual sensor output error injection gain matrix |
| $\boldsymbol{M}$ | Virtual actuator feedback gain matrix |
| $\boldsymbol{N}$ | Virtual actuator feedthrough gain matrix |
| $\boldsymbol{P}$ | Virtual sensor feedthrough gain matrix |
| $\boldsymbol{Q}$ | Weight matrix |
| $\boldsymbol{S}$ | Diagonal matrix of inverse sector bounds |
| $\boldsymbol{T}$ | Similarity transformation matrix |
| $\boldsymbol{T}(s)$ | Transfer function matrix |
| $\boldsymbol{X}, \boldsymbol{Y}$ | Variables of a linear matrix inequality |

**Dynamical systems and dynamical operators**

| | |
|---|---|
| $\Sigma_L$ | Nominal closed-loop system |
| $\Sigma_{Lr}$ | Reconfigured closed-loop system |
| $\Sigma_{LFH}$ | Reconfigured closed-loop system based on the fault-hiding approach |
| $\Sigma_P$ | Nominal plant |
| $\Sigma_{Pr}$ | Reconfigured plant |
| $\Sigma_C$ | Controller |
| $\Sigma_R$ | Reconfiguration block |
| $\Omega_L$ | Nominal closed-loop operator |
| $\Omega_{Lr}$ | Reconfigured closed-loop operator |
| $\Omega_{LFH}$ | Reconfigured closed-loop operator based on the fault-hiding approach |
| $\Omega_P$ | Nominal plant operator |
| $\Omega_{Pr}$ | Reconfigured plant operator |
| $\Omega_C$ | Controller operator |
| $\Omega_R$ | Reconfiguration block operator |

**Sub- and Superscripts**

| | |
|---|---|
| $\overline{(\cdot)}$ | Upper bound on $(\cdot)$ |
| $\underline{(\cdot)}$ | Lower bound on $(\cdot)$ |
| $(\cdot)_f$ | Quantity in a faulty plant |

**Mathematical operators**

| | |
|---|---|
| $\triangleq$ | Equal by definition |
| $\|\cdot\|_p$ | Vector $p$-norm or induced matrix p-norm |
| $\|\cdot\|_{\mathcal{L}_p}$ | $\mathcal{L}_p$-norm of signal |
| $\|\cdot\|_{H_\infty}$ | $H_\infty$-norm of system |
| rank$(\cdot)$ | Rank of a matrix $(\cdot)$ |
| ker$(\cdot)$ | Kernel space of a matrix $(\cdot)$ |
| im$(\cdot)$ | Image space of a matrix $(\cdot)$ |
| span$(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k)$ | Linear space spanned by the vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ |
| $\boldsymbol{A}\|\mathcal{V}$ | Restriction of matrix $\boldsymbol{A}$ to $\boldsymbol{A}$-invariant subspace $\mathcal{V}$ |
| $(\boldsymbol{A})^+$ | Pseudoinverse of matrix $\boldsymbol{A}$ satisfying all four Moore-Penrose equations |
| $(\boldsymbol{A})^\dagger$ | Right inverse of matrix $\boldsymbol{A}$ satisfying $\boldsymbol{A}\boldsymbol{A}^\dagger = \boldsymbol{I}$, $\boldsymbol{A}^\dagger = \boldsymbol{A}^T(\boldsymbol{A}\boldsymbol{A}^T)^{-1}$ |
| $(\boldsymbol{A})^\ddagger$ | Left inverse of matrix $\boldsymbol{A}$ satisfying $\boldsymbol{A}^\ddagger\boldsymbol{A} = \boldsymbol{I}$, $\boldsymbol{A}^\dagger = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T$ |
| $\succ, \prec$ | Positive definite, negative definite |
| $\star$ | Symmetric entry or block in a symmetric matrix |

# Appendix B
# Glossary of Fault-Tolerant Control

The following notions are widely used in the literature on fault-tolerant control. In spite of their wide-spread use, different definitions are available for several of the notions, in particular for dependability. The definitions given here follow the recent monographs on fault-tolerant control [21, 83], but can be traced back to the fields of operations research and dependable computing [79, 164].

**Availability** is the probability that a system is operational when it is needed. Availability is influenced by maintenance strategies, whereas reliability is independent of maintenance strategies.

**Dependability** denotes the combination of safety, reliability, and availability.

**Failures** denote a condition of a system or a component of a system that entirely ceases to perform its function. Failure are special types of faults. Every failure is a fault, but not vice versa.

**Faults** denote a condition of a system or a component of a system that cannot fully perform the function any longer that it is designed for. Fault have location, strength, and temporal behavior. This monograph only concentrates on the location (specific actuators and sensors) and considers faults that remain effective for ever once they have occurred.

**Fault case** denotes a constellation of faults, namely the set of all faults that act on the system.

**Fault detection** denotes the process of deciding whether or not faults are present in the system. Fault detection states neither the locations nor the strengths of the current faults, but only the existence of faults.

**Fault isolation** provides the locations of faults in the system. This monograph assumes that the fault has been uniquely isolated before control reconfiguration activates.

**Fault identification** provides the location and strength of the faults in the system. The reconfigurable control methods shown in this monograph do not require

perfect fault identification due to inherent robustness and the possibility of forcing a shutdown of actuators with uncertain fault status, and the possibility of ignoring the values provided by faulty sensors.

**Fault-tolerant systems** continue to function in spite of faults, possibly with degraded performance. In other words, faults at the component level do not cause a failure at the system level. Fault-tolerance increases a system's reliability and dependability.

**Redundancy** denotes the presence of more than one means for achieving a given goal. Physical redundancy (also called parallel redundancy) refers to the multiple installation of critical components such as actuators or sensor. If one component fails, the other components replace the failed one. Analytical redundancy refers to the presence of functionally similar components, where the functional similarities are expressed in a system model. Adequate utilisation of analytical redundancy usually requires good models that contain a description of the redundancy.

**Reliability** is the probability that a system accomplishes its function for a specified period of time under normal operating conditions. Reliability analysis provides statistical information about the probability of failures, but does not make statements about the system status at any particular time. Fault-tolerant control cannot change the reliability of the system components, but it can improve the overall system reliability.

**Safety** is the absence of danger. System safety is often achieved by implementing a safety system that brings the system into a safe state if critical component faults or failures occur. A system with this property is called a fail-safe system.

# Appendix C
# Linear Subspaces

The notions of controlled (or $(\boldsymbol{A},\boldsymbol{B})$-) invariant subspaces and their duals, the conditioned (or $(\boldsymbol{C},\boldsymbol{A})$-) invariant subspaces [12, 227] are introduced here. For computational aspects, see [70]. They are mostly required for the statement of certain results given in Chapter 4. Consider the system $\Sigma_1 : \dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t)$ with $\boldsymbol{x}(t) \in \mathcal{X} \triangleq \mathbb{R}^n$ and $\boldsymbol{u}(t) \in \mathcal{U} \triangleq \mathbb{R}^m$. An absolutely continuous (a.c.) function $\boldsymbol{x}(t) : \mathbb{R}_+ \to \mathbb{R}^n$ satisfying this differential equation for some $\boldsymbol{u}$ is called an a.c. trajectory of $\Sigma_1$. A subspace $\mathcal{V} \subseteq \mathcal{X}$ is said to be $(\boldsymbol{A},\boldsymbol{B})$-invariant if there exists a matrix $\boldsymbol{F} : \mathcal{X} \to \mathcal{U}$ such that $(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F})\mathcal{V} \subseteq \mathcal{V}$. An equivalent geometric characterisation is that $\mathcal{V}$ must satisfy $\boldsymbol{A}\mathcal{V} \subseteq \mathcal{V} + \operatorname{im}\boldsymbol{B}$. Given a subspace $\mathcal{K} \subset \mathcal{X}$, it can be shown that there exists a unique supremal $(\boldsymbol{A},\boldsymbol{B})$-invariant subspace contained in $\mathcal{K}$, denoted by $\mathcal{V}^*(\mathcal{K})$. If $\mathcal{V}$ is an $\boldsymbol{A}$-invariant subspace, then the notation $\boldsymbol{A}|\mathcal{V}$ denotes the restriction of $\boldsymbol{A}$ to $\mathcal{V}$, defined through the equation $(\boldsymbol{A}|\mathcal{V})\boldsymbol{x} \triangleq \boldsymbol{A}\boldsymbol{x}$ for $\boldsymbol{x} \in \mathcal{V}$. Furthermore, it may be required from $\boldsymbol{F}$ that in addition to rendering $\mathcal{V}^*(\mathcal{K})$ invariant, $\sigma(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F}|\mathcal{V}^*(\mathcal{K}))$ belongs to a prescribed good region $\mathbb{C}_g \subset \mathbb{C}$. A subspace $\mathcal{V}_g \subseteq \mathcal{X}$ is called a stabilisability subspace if there exists a matrix $\boldsymbol{F} : \mathcal{X} \to \mathcal{U}$ such that $(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F})\mathcal{V}_g \subseteq \mathcal{V}_g$ and the closed-loop spectrum lies in a stability set $\mathbb{C}_g$, $\sigma(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F}|\mathcal{V}_g) \subset \mathbb{C}_g$. The supremal stabilisability subspace contained in a given subspace $\mathcal{K} \subset \mathcal{X}$ is well defined and denoted by $\mathcal{V}_g^*(\mathcal{K})$. Furthermore, an $(\boldsymbol{A},\boldsymbol{B})$-invariant subspace $\mathcal{R}$ such that $(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F})\mathcal{R} \subseteq \mathcal{R}$ and $\sigma(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{F}|\mathcal{R})$ is freely assignable is called a controllability subspace. The supremal controllability subspace contained in a subspace $\mathcal{K} \subset \mathcal{X}$ is well-defined and denoted by $\mathcal{R}^*(\mathcal{K})$. Finally, the supremal almost controlled-invariant stabilisability subspace contained in $\mathcal{K}$ is unique, well-defined, and denoted by $\mathcal{V}_{b,g}^*(\mathcal{K})$ [222].

Dual notions are defined for the system $\Sigma_2 : \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$, $\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x}$, with $\boldsymbol{y} \in \mathcal{Y} \triangleq \mathbb{R}^q$. A subspace $\mathcal{S} \subseteq \mathcal{X}$ is called conditioned ($(\boldsymbol{C},\boldsymbol{A})$-) invariant if there exists a matrix $\boldsymbol{J} : \mathcal{Y} \to \mathcal{X}$ such that $(\boldsymbol{A} - \boldsymbol{J}\boldsymbol{C})\mathcal{S} \subseteq \mathcal{S}$ [191], with the equivalent geometric characterisation that $\mathcal{S}$ must satisfy $\boldsymbol{A}(\mathcal{S} \cap \ker\boldsymbol{C}) \subseteq \mathcal{S}$. A $(\boldsymbol{C},\boldsymbol{A})$-invariant subspace $\mathcal{S}_g$ is called a detectability subspace if $\mathcal{S}_g^\perp$ is a stabilisability subspace relative to the pair $(\boldsymbol{A}^T,\boldsymbol{C}^T)$. A $(\boldsymbol{C},\boldsymbol{A})$-invariant subspace $\mathcal{N}$ is called an observability subspace if $\mathcal{N}^\perp$ is a controllability subspace relative to the pair $(\boldsymbol{A}^T,\boldsymbol{C}^T)$. Given a subspace $\mathcal{K}$, it can be shown that $\mathcal{S}^*(\mathcal{K})$, $\mathcal{S}_g^*(\mathcal{K})$, $\mathcal{N}^*(\mathcal{K})$, and $\mathcal{S}_{b,g}^*(\mathcal{K})$ denoting the infimal $(\boldsymbol{C},\boldsymbol{A})$-invariant subspace, the infimal detectability subspace, the infimal observability subspace, and the infimal almost $(\boldsymbol{C},\boldsymbol{A})$-invariant detectability subspace containing $\mathcal{K}$, are well defined and unique [223].

# Appendix D
# Proofs

## D.1 Proofs of Chapter 4

### *D.1.1 Proof of Theorem 4.6 and Theorem 4.13*

The claimed universality is proven by constructing a solution to Problem 3.2. Given a nominal system $\Sigma_P$ and a faulty system $\Sigma_{Pf}$, conditions are sought under which for every stabilizing controller $\Sigma_C$ of the form (4.10), there exists a redesigned controller $\Sigma_{Cr}$ of the form (4.10) such that the nominal closed-loop transfer function from $\boldsymbol{r} \mapsto z$ matches that of the reconfigured closed-loop system from $\boldsymbol{r} \mapsto z_f$. The controller (4.10) is denoted in the Laplace domain by

$$\boldsymbol{u}(s) = \boldsymbol{K}_r(s)\boldsymbol{r}(s) - \boldsymbol{K}_y(s)\boldsymbol{y}(s).$$

It may be assumed that its state-space representation (4.10) is minimal, because any non-minimal elements do not affect the controller I/O behavior. It may be furthermore assumed that the faulty system (4.14) is stabilizable and detectable, because otherwise no stabilizing controller exists.

The nominal system (4.1) has the transfer function

$$\boldsymbol{P}(s) = \begin{pmatrix} \boldsymbol{P}_{\boldsymbol{u}\to\boldsymbol{y}}(s) & \boldsymbol{P}_{\boldsymbol{d}\to\boldsymbol{y}}(s) \\ \boldsymbol{P}_{\boldsymbol{u}\to z}(s) & \boldsymbol{P}_{\boldsymbol{d}\to z}(s) \end{pmatrix}, \tag{D.1}$$

likewise the faulty system (4.14) has the transfer function

$$\boldsymbol{F}(s) = \begin{pmatrix} \boldsymbol{F}_{\boldsymbol{u}_f\to\boldsymbol{y}_f}(s) & \boldsymbol{F}_{\boldsymbol{d}\to\boldsymbol{y}_f}(s) \\ \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s) & \boldsymbol{P}_{\boldsymbol{d}\to z}(s) \end{pmatrix}, \tag{D.2}$$

where due to Definition 4.2 and Definition 4.3, $\boldsymbol{F}_{22}(s) = \boldsymbol{P}_{\boldsymbol{d}\to z}(s)$ is nominal.

Using the nominal controller $\boldsymbol{K}(s) \triangleq [\boldsymbol{K}_r(s) \ -\boldsymbol{K}_y(s)]$, where $\boldsymbol{K}_r(s)$ must be stable, straightforward calculations yield the following transfer functions $\boldsymbol{T}_{\boldsymbol{r}\to z}(s)$ and $\boldsymbol{T}_{\boldsymbol{d}\to z}(s)$ for the nominal closed-loop system $(\Sigma_P, \Sigma_C)$

$$\boldsymbol{T}_{\boldsymbol{r}\to z}(s) = \boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{K}_r(s) - \boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{K}_y(s)\left(\boldsymbol{I} + \boldsymbol{P}_{\boldsymbol{u}\to \boldsymbol{y}}(s)\boldsymbol{K}_y(s)\right)^{-1}\boldsymbol{P}_{\boldsymbol{u}\to \boldsymbol{y}}(s)\boldsymbol{K}_r(s) \tag{D.3}$$

$$\boldsymbol{T}_{\boldsymbol{d}\to z}(s) = \boldsymbol{P}_{\boldsymbol{d}\to z}(s) - \boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{K}_y(s)\left(\boldsymbol{I} + \boldsymbol{P}_{\boldsymbol{u}\to \boldsymbol{y}}(s)\boldsymbol{K}_y(s)\right)^{-1}\boldsymbol{P}_{\boldsymbol{d}\to \boldsymbol{y}}(s). \tag{D.4}$$

With a reconfigured controller $\Sigma_{Cr} : \boldsymbol{K}'(s) \triangleq [\boldsymbol{K}'_r(s) \; -\boldsymbol{K}'_y(s)]$, corresponding transfer functions for the reconfigured closed-loop system $(\Sigma_{Pf}, \Sigma_{Cr})$ are obtained:

$$\boldsymbol{T}'_{\boldsymbol{r}\to z_f}(s) = \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{K}'_r(s) - \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{K}'_y(s)\left(\boldsymbol{I} + \boldsymbol{F}_{\boldsymbol{u}_f\to \boldsymbol{y}_f}(s)\boldsymbol{K}'_y(s)\right)^{-1}\boldsymbol{F}_{\boldsymbol{u}_f\to \boldsymbol{y}_f}(s)\boldsymbol{K}'_r(s) \tag{D.5}$$

$$\boldsymbol{T}'_{\boldsymbol{d}\to z_f}(s) = \boldsymbol{P}_{\boldsymbol{d}\to z}(s) - \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{K}'_y(s)\left(\boldsymbol{I} + \boldsymbol{F}_{\boldsymbol{u}_f\to \boldsymbol{y}_f}(s)\boldsymbol{K}'_y(s)\right)^{-1}\boldsymbol{F}_{\boldsymbol{d}\to \boldsymbol{y}_f}(s). \tag{D.6}$$

Due to the results by Youla and Kucera [193, 245, 246], all stabilizing nominal feedback controllers $\boldsymbol{K}_y(s)$ are parameterized by the stable transfer function $\boldsymbol{Q}(s)$, and all stabilizing reconfigured feedback controllers $\boldsymbol{K}'_y(s)$ are parameterized over the stable transfer function $\boldsymbol{Q}'(s)$ by means of the relations

$$\boldsymbol{Q}(s) = \boldsymbol{K}_y(s)\left(\boldsymbol{I} + \boldsymbol{P}_{\boldsymbol{u}\to \boldsymbol{y}}(s)\boldsymbol{K}_y(s)\right)^{-1} \tag{D.7}$$

$$\boldsymbol{Q}'(s) = \boldsymbol{K}'_y(s)\left(\boldsymbol{I} + \boldsymbol{F}_{\boldsymbol{u}_f\to \boldsymbol{y}_f}(s)\boldsymbol{K}'_y(s)\right)^{-1}. \tag{D.8}$$

Problem 3.2 is solved if and only if for all $(\Sigma_P, \Sigma_C)$-stabilizing transfer function $\boldsymbol{K}(s)$, there exists a $(\Sigma_{Pf}, \Sigma_{Cr})$-stabilizing transfer function $\boldsymbol{K}'(s)$ such that

$$\boldsymbol{T}'_{\boldsymbol{r}\to z_f}(s) = \boldsymbol{T}_{\boldsymbol{r}\to z}(s) \tag{D.9}$$

$$\boldsymbol{T}'_{\boldsymbol{d}\to z_f}(s) = \boldsymbol{T}_{\boldsymbol{d}\to z}(s). \tag{D.10}$$

Using the Youla-parameters $\boldsymbol{Q}(s)$ and $\boldsymbol{Q}'(s)$ to characterize the feedback, the equivalent criteria are that for all stable $(\boldsymbol{K}_r(s), \boldsymbol{Q}(s))$, there exist stable $(\boldsymbol{K}'_r(s), \boldsymbol{Q}'(s))$ such that

$$\boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{K}_r(s) = \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{K}'_r(s) \tag{D.11}$$

$$\boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{Q}(s)\boldsymbol{P}_{\boldsymbol{u}\to \boldsymbol{y}}(s)\boldsymbol{K}_r(s) = \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{Q}'(s)\boldsymbol{F}_{\boldsymbol{u}_f\to \boldsymbol{y}_f}(s)\boldsymbol{K}'_r(s) \tag{D.12}$$

$$\boldsymbol{P}_{\boldsymbol{u}\to z}(s)\boldsymbol{Q}(s)\boldsymbol{P}_{\boldsymbol{d}\to \boldsymbol{y}}(s) = \boldsymbol{F}_{\boldsymbol{u}_f\to z_f}(s)\boldsymbol{Q}'(s)\boldsymbol{F}_{\boldsymbol{d}\to \boldsymbol{y}_f}(s). \tag{D.13}$$

(*Actuator faults*) Now consider the case of pure actuator faults ($\boldsymbol{C}_f = \boldsymbol{C}$). It follows that $\boldsymbol{F}_{\boldsymbol{d}\to \boldsymbol{y}_f}(s) = \boldsymbol{P}_{\boldsymbol{d}\to \boldsymbol{y}}(s)$, and the solvability of (D.11) for $\boldsymbol{K}'_r(s)$ implies the solvability of (D.13) for $\boldsymbol{Q}'(s)$. Expansion of (D.11) leads to

$$\boldsymbol{C}_z(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}\boldsymbol{K}_r(s) = \boldsymbol{C}_z(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}_f\boldsymbol{K}'_r(s), \tag{D.14}$$

for which stable solutions $\boldsymbol{K}'_r(s)$ for all stable $\boldsymbol{K}_r(s)$ are sought. In other words, one ranges over the entire set of stabilizing controllers to achieve $\boldsymbol{z}_\Delta \triangleq \boldsymbol{z}(t) - \boldsymbol{z}_f(t) = \boldsymbol{0}\ \forall t \in \mathbb{R}_+$. Defining $\boldsymbol{x}_\omega(t) \triangleq \boldsymbol{x}(t) - \boldsymbol{x}_f(t)$, the dynamics for $\boldsymbol{z}_\Delta$ are governed by

$$\begin{cases} \dot{\boldsymbol{x}}_\omega(t) &= \boldsymbol{A}\boldsymbol{x}_\omega(t) + \boldsymbol{B}\boldsymbol{u}_c(t) - \boldsymbol{B}_f\boldsymbol{u}_f(t),\ \boldsymbol{x}_\omega(0) = \boldsymbol{0} \\ \boldsymbol{z}_\Delta(t) &= \boldsymbol{C}_z\boldsymbol{x}_\omega(t). \end{cases}$$

Finding a control $\boldsymbol{u}_f(t)$ such that for all stabilizing measurable controls $\boldsymbol{u}(t)$ the output $\boldsymbol{z}_\Delta$ is measurable is therefore equivalent to a known disturbance decoupling problem with stability (DDPS'), which is known to be solvable if and only if Condition (4.55) is satisfied. The same holds for the almost-trajectory recovery and Condition (4.56). Equation (D.12) is solvable for all stable $\boldsymbol{Q}$ if Equation (D.13) is solvable. Therefore, a suitable virtual actuator exists, and the correct initial condition is $\boldsymbol{\zeta}_0 = \boldsymbol{0}$.

(*Sensor faults*) Now consider the case of pure sensor faults ($\boldsymbol{B}_f = \boldsymbol{B}$). It follows that $\boldsymbol{F}_{\boldsymbol{u}_f \to \boldsymbol{z}_f}(s) = \boldsymbol{P}_{\boldsymbol{u}\to\boldsymbol{z}}(s)$, and Condition (D.11) is automatically satisfied for $\boldsymbol{K}_r'(s) = \boldsymbol{K}_r(s)$. Equation (D.12) is easily satisfiable by choosing $\boldsymbol{Q}'(s)$ as an observer for the faulty system. Further conditions on the observer are obtained from Equation (D.13), which is solvable in $\boldsymbol{Q}'(s)$ for fixed $\boldsymbol{Q}(s)$ iff [223, Appendix B] both following equations are solvable in $\boldsymbol{Q}_1'(s)$, $\boldsymbol{Q}_2'(s)$:

$$\boldsymbol{P}_{\boldsymbol{u}\to\boldsymbol{z}}(s)\boldsymbol{Q}(s) = \boldsymbol{P}_{\boldsymbol{u}\to\boldsymbol{z}}(s)\boldsymbol{Q}_1'(s) \tag{D.15}$$

$$\boldsymbol{Q}(s)\boldsymbol{P}_{\boldsymbol{d}\to\boldsymbol{y}}(s) = \boldsymbol{Q}_2'(s)\boldsymbol{F}_{\boldsymbol{d}\to\boldsymbol{y}_f}(s). \tag{D.16}$$

Equation (D.15) is trivially satisfiable by $\boldsymbol{Q}_1(s) = \boldsymbol{Q}(s)$ for all $\boldsymbol{Q}(s)$. It remains to solve Equation (D.16) for all stable $\boldsymbol{Q}(s)$, which is after expansion

$$\boldsymbol{Q}(s)\boldsymbol{C}(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}_{\boldsymbol{d}} = \boldsymbol{Q}_2(s)\boldsymbol{C}_f(s\boldsymbol{I} - \boldsymbol{A})^{-1}\boldsymbol{B}_{\boldsymbol{d}},$$

and for which we seek stable solutions $\boldsymbol{Q}_2'(s)$ for all stable $\boldsymbol{Q}(s)$. In a way dual to the actuator case, this problem is equivalent to a decoupled-estimation problem with measurement throughput and stability (DDEPS'), which is known to be solvable if and only if Condition (4.33) is satisfied. The same holds for the almost-trajectory recovery and Condition (4.34). A virtual sensor exists that solves the problem, and $\boldsymbol{\zeta}_0 = \boldsymbol{x}_0$.

## D.2 Proofs of Chapter 6

### *D.2.1 Proof of Theorem 6.1*

Closed-loop stability is established based on the weak fault-hiding property (Lemma 6.1), the nominal closed-loop IOS (Assumption 5.3), and the established stability properties of the observation error and difference systems together with Proposition 2.2. The equivalent closed-loop block diagram can be drawn as in Fig. D.2 due to Equation (6.3) and interpreted as the cascade interconnection of the nominal closed-loop system with a difference and error subsystem due to Lemma 6.1.

First, consider the feedback connection $(\Sigma_e, \Sigma_\Delta)$ (shaded block in Fig. D.1), to which the signals $\boldsymbol{u}_c$, $\boldsymbol{d}$, and $\tilde{\boldsymbol{x}}$ are external inputs. Normally, to conclude ISS of feedback connections, a small-gain argument is needed. In this case, however, $\Sigma_e$

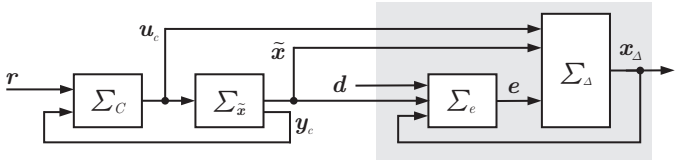


**Fig. D.1** Transformed reconfigured closed-loop system (5.8), (5.4), (6.1), (6.2).

is exponentially stable for arbitrary interconnection inputs $\tilde{\boldsymbol{x}}$, $\boldsymbol{x}_\Delta$, which is used as follows. From Lemma 6.2, the observation error dynamics is 0-GES for arbitrary inputs $\tilde{\boldsymbol{x}}$ and $\boldsymbol{x}_\Delta$ and ISS w.r.t. the disturbance $\boldsymbol{d}$. In other words,

$$\|\boldsymbol{e}(t)\| \le \beta_e(\|\boldsymbol{e}(s)\|, t-s) + \gamma_d(\sup_{s\le\tau\le t}\|\boldsymbol{d}(\tau)\|) \tag{D.17}$$

for some $\beta_e \in \mathcal{KL}$, $\gamma_d \in \mathcal{K}$. The difference system is globally exponentially stable with $\boldsymbol{u}_c = \boldsymbol{e} = \boldsymbol{0}$ for arbitrary $\tilde{\boldsymbol{x}}$ and ISS w.r.t. $\boldsymbol{u}_c$ and $\boldsymbol{e}$, hence for $t \ge s \ge t_0 \ge 0$ it follows that

$$\|\boldsymbol{x}_\Delta(t)\| \le \beta_\Delta(\|\boldsymbol{x}_\Delta(s)\|, t-s) + \gamma_u\left(\sup_{s\le\tau\le t}\|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{s\le\tau\le t}\|\boldsymbol{e}(\tau)\|\right) \tag{D.18}$$

for some $\beta_\Delta \in \mathcal{KL}$, $\gamma_u, \gamma_e \in \mathcal{K}$. Applying (D.18) with $s = (t+t_0)/2$ gives

$$\|\boldsymbol{x}_\Delta(t)\| \le \beta_\Delta\left(\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\|, \frac{t-t_0}{2}\right) + \gamma_u\left(\sup_{\frac{t+t_0}{2}\le\tau\le t}\|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{\frac{t+t_0}{2}\le\tau\le t}\|\boldsymbol{e}(\tau)\|\right). \tag{D.19}$$

To estimate the term $\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\|$, apply (D.18) with $s = t_0$ and $t$ is replaced by $(t+t_0)/2$ to get

$$\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\| \le \beta_\Delta\left(\|\boldsymbol{x}_\Delta(t_0)\|, \frac{t-t_0}{2}\right) + \gamma_u\left(\sup_{t_0\le\tau\le\frac{t+t_0}{2}}\|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{t_0\le\tau\le\frac{t+t_0}{2}}\|\boldsymbol{e}(\tau)\|\right). \tag{D.20}$$

To estimate the term $\sup_{t_0\le\tau\le\frac{t+t_0}{2}}\|\boldsymbol{e}(\tau)\|$, use (D.17) to obtain

$$\sup_{t_0\le\tau\le\frac{t+t_0}{2}}\|\boldsymbol{e}(\tau)\| \le \beta_e(\|\boldsymbol{e}(t_0)\|, 0) + \gamma_d\left(\sup_{t_0\le\tau\le\frac{t+t_0}{2}}\|\boldsymbol{d}(\tau)\|\right) \tag{D.21}$$

$$\sup_{\frac{t+t_0}{2}\le\tau\le t}\|\boldsymbol{e}(\tau)\| \le \beta_e\left(\left\|\boldsymbol{e}\left(\frac{t+t_0}{2}\right)\right\|, \frac{t-t_0}{2}\right) + \gamma_d\left(\sup_{\frac{t+t_0}{2}\le\tau\le t}\|\boldsymbol{d}(\tau)\|\right)$$

$$\leq \beta_e\left(\|\boldsymbol{e}(t_0)\|, \frac{t-t_0}{2}\right) + \gamma_d\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{d}(\tau)\|\right) \tag{D.22}$$

By inserting (D.20)–(D.22) into (D.19) and using the fact that for $\gamma \in \mathcal{K}$ the relation $\gamma(r_1+r_2) \leq \gamma(2r_1)+\gamma(2r_2)$, the inequality

$$\begin{aligned}
\|\boldsymbol{x}_\Delta(t)\| \leq{}& \beta_\Delta\left(2\beta_\Delta\left(\|\boldsymbol{x}_\Delta(t_0)\|, \frac{t-t_0}{2}\right), \frac{t-t_0}{2}\right) + \beta_\Delta\left(2\gamma_e\left(2\beta_e\left(\|\boldsymbol{e}(t_0)\|, 0\right)\right), \frac{t-t_0}{2}\right) \\
&+ \gamma_e\left(2\beta_e\left(\|\boldsymbol{e}(t_0)\|, \frac{t-t_0}{2}\right)\right) + \beta_\Delta\left(2\gamma_u\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{u}_c(\tau)\|\right), \frac{t-t_0}{2}\right) \\
&+ \beta_\Delta\left(2\gamma_e\left(2\gamma_d\left(\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{d}(\tau)\|\right)\right), \frac{t-t_0}{2}\right) \\
&+ \gamma_u\left(\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(2\gamma_d\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{d}(\tau)\|\right)\right).
\end{aligned} \tag{D.23}$$

follows. Clearly, the sum of the first five terms is in class $\mathcal{KL}$, whereas the sum of the last two terms is in class $\mathcal{K}$. Therefore the following shorter notation is introduced,

$$\|\boldsymbol{x}_\Delta(t)\| \leq \beta_1(\|\boldsymbol{x}_\Delta(t_0)\|, t-t_0) + \beta_2(\|\boldsymbol{e}(t_0)\|, t-t_0) + \gamma_1(\|\boldsymbol{u}_c\|_{\mathcal{L}_\infty}) + \gamma_2(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}), \tag{D.24}$$

where $\beta_1(\cdot) = \beta_\Delta \circ 2\beta_\Delta(\cdot) \in \mathcal{KL}$ and $\beta_2(\cdot) = \beta_\Delta \circ 2\gamma_e \circ 2\beta_e(\cdot) + \gamma_e \circ 2\beta_e(\cdot) \in \mathcal{KL}$ as well as $\gamma_1(\cdot) = \gamma_u(\cdot) + \beta_\Delta \circ 2\gamma_u(\cdot) \in \mathcal{K}$ and $\gamma_2(\cdot) = \gamma_e \circ 2\gamma_d(\cdot) + \beta_\Delta \circ 2\gamma_e \circ 2\gamma_d(\cdot) \in \mathcal{K}$.

The interconnection $(\Sigma_e, \Sigma_\Delta)$ satisfies the following relation, obtained by observing that $\|(\boldsymbol{e}^T, \boldsymbol{x}_\Delta^T)^T\| \leq \|\boldsymbol{e}\| + \|\boldsymbol{x}_\Delta\|$ and $\|(\boldsymbol{u}_c^T, \boldsymbol{d}^T)^T\| \leq \|\boldsymbol{u}_c\| + \|\boldsymbol{d}\|$ and inserting the above relations:

$$\begin{aligned}
\left\|\begin{pmatrix}\boldsymbol{e}(t)\\ \boldsymbol{x}_\Delta(t)\end{pmatrix}\right\| \leq{}& \beta_e(\|\boldsymbol{e}(t_0)\|, t-t_0) + \gamma_d(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}) + \beta_1(\|\boldsymbol{x}_\Delta(t_0)\|, t-t_0) + \beta_2(\|\boldsymbol{e}(t_0)\|, t-t_0) \\
&+ \gamma_1(\|\boldsymbol{u}_c\|_{\mathcal{L}_\infty}) + \gamma_2(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}) \\
\leq{}& \beta_e\left(\left\|\begin{pmatrix}\boldsymbol{x}_\Delta(t_0)\\ \boldsymbol{e}(t_0)\end{pmatrix}\right\|, t-t_0\right) + \beta_1\left(\left\|\begin{pmatrix}\boldsymbol{x}_\Delta(t_0)\\ \boldsymbol{e}(t_0)\end{pmatrix}\right\|, t-t_0\right) + \beta_2\left(\left\|\begin{pmatrix}\boldsymbol{x}_\Delta(t_0)\\ \boldsymbol{e}(t_0)\end{pmatrix}\right\|, t-t_0\right) \\
&+ \gamma_d\left(\left\|\begin{pmatrix}\boldsymbol{u}_c\\ \boldsymbol{d}\end{pmatrix}\right\|_{\mathcal{L}_\infty}\right) + \gamma_1\left(\left\|\begin{pmatrix}\boldsymbol{u}_c\\ \boldsymbol{d}\end{pmatrix}\right\|_{\mathcal{L}_\infty}\right) + \gamma_2\left(\left\|\begin{pmatrix}\boldsymbol{u}_c\\ \boldsymbol{d}\end{pmatrix}\right\|_{\mathcal{L}_\infty}\right).
\end{aligned}$$

An ISS-characterisation for the interconnected system $(\Sigma_e, \Sigma_\Delta)$ has been obtained, where clearly $[\beta_e(r,s)+\beta_1(r,s)+\beta_2(r,s)] \in \mathcal{KL}$ and $[\gamma_d(r)+\gamma_1(r)+\gamma_2(r)] \in \mathcal{K}$ hold. It is concluded that the subsystem $(\Sigma_e, \Sigma_\Delta)$ is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{d}, \tilde{\boldsymbol{x}})$, hence also IOS w.r.t. the outputs $(\boldsymbol{e}, \boldsymbol{x}_\Delta)$. The system $(\Sigma_{\tilde{P}}, \Sigma_C)$ is IOS w.r.t. the input $\boldsymbol{r}$ and the output $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}})$ by Assumption 5.3. The series connection $((\Sigma_{\tilde{P}}, \Sigma_C), (\Sigma_e, \Sigma_\Delta))$ that represents the reconfigured closed-loop system is IOS by Proposition 2.2, which completes the proof.

### D.2.2 Proof of Theorem 6.2

Consider the LMI (6.8) for characterising stabilising gains of the Hammerstein-Wiener virtual sensor. The substitutions $\boldsymbol{C}_f \to -\boldsymbol{B}_f^T$, $\boldsymbol{A} \to \boldsymbol{A}^T$, and $\boldsymbol{S}_h \to \boldsymbol{S}_\varphi$ transform the LMI (6.8) to the LMI

$$\left(\begin{array}{c|c} -(\boldsymbol{A}\boldsymbol{X}_s + \boldsymbol{X}_s\boldsymbol{A}^T) & -\boldsymbol{B}_f - \boldsymbol{Y}_s \\ \hline \star & 2\boldsymbol{S}_\varphi \end{array}\right) \succ 0.$$

By relabeling $\boldsymbol{X}_a \triangleq \boldsymbol{X}_s$ and $\boldsymbol{Y}_a \triangleq -\boldsymbol{Y}_s^T$, the LMI (6.11) has been derived. This result implies that any solution $(\boldsymbol{X}_s, \boldsymbol{Y}_s)$ to the LMI (6.8) with the parameters $(\boldsymbol{A}, \boldsymbol{C}_f)$ corresponding to a Hammerstein-Wiener system subject to sensor faults also provides a solution $(\boldsymbol{X}_a, \boldsymbol{Y}_a)$ to the LMI (6.11) with the parameters $(\boldsymbol{A}, \boldsymbol{B}_f)$, which corresponds to a Hammerstein-Wiener system subject to actuator faults. The solutions $\boldsymbol{L}$ and $\boldsymbol{M}$ are linked by the relation

$$\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1} = -\boldsymbol{Y}_s^T\boldsymbol{X}_s^{-1} = -(\boldsymbol{X}_s^{-1}\boldsymbol{Y}_s)^T = -\boldsymbol{L}^T,$$

which completes the proof.

### D.2.3 Proof of Theorem 6.4

It is shown that the problem of finding a new stabilising state-feedback law $\boldsymbol{u}_f(t) = \boldsymbol{K}\boldsymbol{x}_f(t)$ is equivalent to finding a stabilising Hammerstein virtual actuator (6.17). The combination of the new state-feedback controller with the faulty plant (5.8) leads to the reconfigured closed-loop system $\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{B}_f\boldsymbol{\varphi}_f\left(\boldsymbol{K}\boldsymbol{x}_f(t)\right) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)$, which has the same form as the difference system (6.21) after the substitutions $\boldsymbol{M} = \boldsymbol{K}$, $\boldsymbol{x}_\Delta = \boldsymbol{x}_f$ up to a sign and differing exogenous inputs. Thus, the problems are equivalent, and the virtual actuator is state-feedback universal in the sense of Definition 6.4.

### D.2.4 Proof of Theorem 6.5

From Theorem 2.5, it follows directly that the observation error (6.29) is asymptotically stable for $\boldsymbol{d} \equiv \boldsymbol{0}$. In fact, the unforced observation error is even globally uniformly exponentially stable [97, Proof of Theorem 7.1].

From the exponential stability of the unforced observation error and from Theorem 2.1, it follows that the unforced observation error has a Lyapunov function $V(\boldsymbol{e})$. The derivative of $V(\boldsymbol{e})$ with respect to the forced observation error system (6.5) satisfies the relations

$$\begin{aligned}\dot{V}(\boldsymbol{e}) &= \nabla V(\boldsymbol{e})\Big(\boldsymbol{A}\boldsymbol{e}(t) + \boldsymbol{L}\Big(\boldsymbol{h}_f(\boldsymbol{C}_f\boldsymbol{x}_f(t)) - \hat{\boldsymbol{h}}_f(\hat{\boldsymbol{C}}_f(\boldsymbol{x}_f(t) + \boldsymbol{e}(t)))\Big) - \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)\Big) \\ &\leq -c_3\|\boldsymbol{e}\|^2 + c_4(\hat{R}_0 + \hat{R}\|\boldsymbol{e}\|)\|\boldsymbol{e}\| \cdot \|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\| + c_4\|\boldsymbol{e}\| \cdot \|\boldsymbol{B}_{\boldsymbol{d}}\| \cdot \|\boldsymbol{d}\|\end{aligned}$$

due to the robust Lipschitz condition (6.30) and Theorem 2.1. The first term $-c_3\|\boldsymbol{e}\|^2$ is obviously always negative. Using the parameter $\theta \in (0,1)$, the inequality is rewritten as

$$\begin{aligned}\dot{V}(\boldsymbol{e}) &\leq -c_3(1-\theta)\|\boldsymbol{e}\|^2 - c_3\theta\|\boldsymbol{e}\|^2 + c_4(\hat{R}_0 + \hat{R}\|\boldsymbol{e}\|)\|\boldsymbol{e}\| \cdot \|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\| + c_4\|\boldsymbol{e}\| \cdot \|\boldsymbol{B}_{\boldsymbol{d}}\| \cdot \|\boldsymbol{d}\| \\ &\leq -(1-\theta)c_3\|\boldsymbol{e}\|^2, \quad \text{if } \|\boldsymbol{e}\| \geq \frac{\hat{R}_0}{\hat{R}} + \frac{c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|}{\theta c_3 - c_4\hat{R}\|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\|}\|\boldsymbol{d}\|.\end{aligned}$$

Hence, Theorem 2.2 is invoked with $\alpha_1(r) = c_1 r^2$, $\alpha_2(r) = c_2 r^2$, $W(\boldsymbol{e}) = (1-\theta)c_3\|\boldsymbol{e}\|^2$ and $\rho(r) = (c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|/(c_3\theta - c_4\hat{R}\|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\|))r$, and the observation error system (6.29) is globally ISpS w.r.t. the input $\boldsymbol{d}$ with $\gamma(r) = \sqrt{c_2/c_1}(c_4\|\boldsymbol{B}_{\boldsymbol{d}}\|/(c_3\theta - c_4\hat{R}\|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\|))r$ for $c_3\theta - c_4\hat{R}\|\boldsymbol{C}_f\| \cdot \|\boldsymbol{L}\| > 0$. The remaining proof of closed-loop stability is identical to the proof of Theorem 6.1, which completes the proof of Theorem 6.5.

### *D.2.5 Proof of Theorem 6.6*

It is shown that the problem of finding a stabilising output-injection control law $\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}\boldsymbol{x}_f + \boldsymbol{K}\boldsymbol{y}_f(t)$ is equivalent to finding a stabilising Hammerstein-Wiener virtual sensor (6.1). The combination of the output-injection control law with the faulty plant (5.8) and nominal controller leads to the reconfigured closed-loop system $\dot{\boldsymbol{x}}_f(t) = \boldsymbol{A}\boldsymbol{x}_f(t) + \boldsymbol{K}\boldsymbol{h}\left(\boldsymbol{C}\boldsymbol{x}_f(t)\right) + \boldsymbol{B}_{\boldsymbol{d}}\boldsymbol{d}(t)$, which has the same basic form as the observation error system (6.26) after the substitutions $\boldsymbol{L} = \boldsymbol{K}$, $\boldsymbol{e} = \boldsymbol{x}_f$ up to a sign and differing exogenous inputs. Thus, the problems are equivalent, and the Hammerstein-Wiener virtual sensor is output-injection universal in the sense of Definition 6.5.

## D.3 Proofs of Chapter 7

### *D.3.1 Proof of Theorem 7.1*

The condition $\bar{\boldsymbol{r}} \in \mathcal{Z}_a$ ensures that the chosen setpoint is an output equilibrium of the faulty saturated system that corresponds to an equilibrium of the virtual actuator $\bar{\boldsymbol{x}}_{\Delta}$ and an input equilibrium $\bar{\boldsymbol{u}} \in \mathcal{U}_c$. It remains to be shown that the equilibrium $\bar{\boldsymbol{r}}$ is always reached from any other feasible output equilibrium. Let $(\bar{\boldsymbol{x}}_{\Delta,1}, \bar{\boldsymbol{u}}_1)$ and $(\bar{\boldsymbol{x}}_{\Delta,2}, \bar{\boldsymbol{u}}_2)$ be two feasible state equilibria. It must be shown that there exists a continuous control input $\boldsymbol{u}_c(t)$ that steers the system from $(\bar{\boldsymbol{x}}_{\Delta,1}, \bar{\boldsymbol{u}}_1)$ to $(\bar{\boldsymbol{x}}_{\Delta,2}, \bar{\boldsymbol{u}}_2)$.

Using the abbreviations $\boldsymbol{A}_{\Delta} = \boldsymbol{A} - \boldsymbol{B}_f\boldsymbol{M}$ and $\boldsymbol{B}_{\Delta} = \boldsymbol{B} - \boldsymbol{B}_f\boldsymbol{N}$, the equilibria are characterised by the equations

$$\begin{aligned}\boldsymbol{0} &= \boldsymbol{A}\bar{\boldsymbol{x}}_{\Delta,1} - \boldsymbol{B}_f \operatorname{sat}\left(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{M}\bar{\boldsymbol{x}}_{\Delta,1} + \boldsymbol{N}\bar{\boldsymbol{u}}_1\right) + \boldsymbol{B} \operatorname{sat}\left(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \bar{\boldsymbol{u}}_1\right) \\ \boldsymbol{0} &= \boldsymbol{A}\bar{\boldsymbol{x}}_{\Delta,2} - \boldsymbol{B}_f \operatorname{sat}\left(\underline{\boldsymbol{u}}_f, \overline{\boldsymbol{u}}_f, \boldsymbol{M}\bar{\boldsymbol{x}}_{\Delta,2} + \boldsymbol{N}\bar{\boldsymbol{u}}_2\right) + \boldsymbol{B} \operatorname{sat}\left(\underline{\boldsymbol{u}}, \overline{\boldsymbol{u}}, \bar{\boldsymbol{u}}_2\right) \\ \bar{\boldsymbol{u}}_1 &= -\boldsymbol{B}_{\Delta}^{+}\boldsymbol{A}_{\Delta}\bar{\boldsymbol{x}}_{\Delta,1}, \; \bar{\boldsymbol{u}}_2 = -\boldsymbol{B}_{\Delta}^{+}\boldsymbol{A}_{\Delta}\bar{\boldsymbol{x}}_{\Delta,2}.\end{aligned}$$

Due to the satisfaction of the strong fault-hiding goal, and the assumption that the nominal closed-loop system tracks feasible setpoints, the input $\boldsymbol{u}_c$ certainly approaches an equilibrium value $\bar{\boldsymbol{u}}_c$, starting from any initial condition $\bar{\boldsymbol{x}}_{\Delta,1}$. The dynamical properties of setpoint change are next studied by showing that the system is incrementally stable [5]. The following relation is obtained from Theorem 2.1 in the same way as in the proof of Theorem 6.1:

$$\dot{V}(\boldsymbol{x}_\Delta - \bar{\boldsymbol{x}}_{\Delta,2}) \leq -c_3(1-\theta)\|\boldsymbol{x}_\Delta - \bar{\boldsymbol{x}}_{\Delta,2}\|^2 \quad \text{for } \|\boldsymbol{x}_\Delta - \bar{\boldsymbol{x}}_{\Delta,2}\| \geq \frac{c_4 L}{c_3 \theta}\|\boldsymbol{u}_c - \bar{\boldsymbol{u}}_2\|.$$

Since it is known that $\boldsymbol{u}_c(t) \to \bar{\boldsymbol{u}}_2$ for $t \to \infty$, the derivative $\dot{V}$ is guaranteed to be eventually negative for $\boldsymbol{x}_\Delta \neq \bar{\boldsymbol{x}}_{\Delta,2}$. Hence, the increment $\boldsymbol{x}_\Delta - \bar{\boldsymbol{x}}_{\Delta,2}$ has been shown to asymptotically vanish, and the state equilibrium is asymptotically reached.

Every admissible setpoint in $\bar{\boldsymbol{z}} \in \mathcal{Z}_a$ is reached by applying a suitable input $\bar{\boldsymbol{u}}_c$. The mapping from $\boldsymbol{u}$ to $\boldsymbol{z}$ is surjective, but not injective. Hence, $\bar{\boldsymbol{u}}_c$ is not unique. In particular, in the linear case multiple input equilibria $\bar{\boldsymbol{u}}_1$, $\bar{\boldsymbol{u}}_2$ … may lead to the same given output equilibrium $\bar{\boldsymbol{z}}$:

$$\boldsymbol{C}_z \boldsymbol{A}^{-1} \boldsymbol{B} \bar{\boldsymbol{u}}_1 = \boldsymbol{C}_z \boldsymbol{A}^{-1} \boldsymbol{B} \bar{\boldsymbol{u}}_2 = \bar{\boldsymbol{z}}.$$

However, not all candidate inputs generating $\bar{\boldsymbol{z}}$ are admissible, because some of them may activate saturations; for example, $\neg(\bar{\boldsymbol{u}}_1 \in \mathcal{U})$ but $\bar{\boldsymbol{u}}_2 \in \mathcal{U}$. The effect occurs particularly when actuator equilibrium values are chosen close to an actuation range boundary. The proper choice of $\boldsymbol{N}$ depends on the reference equilibrium value. From a practical point of view, it is, however, undesirable to change $\boldsymbol{N}$ in response to setpoint changes. This potential problem is due to ambiguity in the solution of the equation $\boldsymbol{C}_z\left(\boldsymbol{A} - \boldsymbol{B}_f \boldsymbol{M}\right)^{-1} \boldsymbol{B} = \boldsymbol{C}_z\left(\boldsymbol{A} - \boldsymbol{B}_f \boldsymbol{M}\right)^{-1} \boldsymbol{B}_f \boldsymbol{N}$, which cannot occur if the faulty input matrix is minimum equilibrium-preserving. Since the output equilibrium $\bar{\boldsymbol{r}}$ is feasible, it follows that $\boldsymbol{C}_z \bar{\boldsymbol{x}}_\Delta = \boldsymbol{0}$, and $\bar{\boldsymbol{z}}_\Delta = \boldsymbol{0}$.

## D.4 Proofs of Chapter 8

### *D.4.1 Proof of Theorem 8.1*

The first synthesis LMI is obtained by inserting the matrix parameters of the transfer function $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_\Delta}(s)$ defined in Equation (8.2) into the LMI of Theorem 4.1 and performing the substitutions $\boldsymbol{X}_{az} = \boldsymbol{X}$ and $\boldsymbol{Y}_{az} = \boldsymbol{M}\boldsymbol{X}_{az}$. The inverse $\boldsymbol{X}_{az}^{-1}$ always exists, since $\boldsymbol{X}_{az}$ is positive definite by requirement, which completes the proof.

### *D.4.2 Proof of Theorem 8.2*

The first synthesis LMI is obtained by inserting the matrix parameters of the transfer function $\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}(s)$ defined in Equation (8.3) into the LMI of Theorem 4.1 and

performing the substitutions $\boldsymbol{X}_{au} = \boldsymbol{X}$ and $\boldsymbol{Y}_{au} = \boldsymbol{M}\boldsymbol{X}_{au}$. The inverse $\boldsymbol{X}_{au}^{-1}$ always exists, since $\boldsymbol{X}_{au}$ is positive definite by requirement, which completes the proof.

### *D.4.3 Proof of Theorem 8.3*

Clearly, the LMI (8.4) characterises the performance loss, and the LMI (8.5) characterises the input amplification. (see Theorem 4.1.) It remains to combine both goals into a single optimisation problem. Note that in both LMI, the virtual actuator gain $\boldsymbol{M}$ is linked to the solutions $\boldsymbol{X}_{az}, \boldsymbol{Y}_{az}$ and $\boldsymbol{X}_{au}, \boldsymbol{Y}_{au}$ in the same way: $\boldsymbol{M} = \boldsymbol{Y}_{az}\boldsymbol{X}_{az}^{-1}$ and $\boldsymbol{M} = \boldsymbol{Y}_{au}\boldsymbol{X}_{au}^{-1}$. From the point of view of independent descriptions of the matrix norms $\|\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{z}_\Delta}\|_{H_\infty}$ and $\|\boldsymbol{T}_{\boldsymbol{u}_c \to \boldsymbol{u}_f}\|_{H_\infty}$, the independent LMI characterisations (8.4), (8.5) with separate variable pairs $(\boldsymbol{X}_{az}, \boldsymbol{Y}_{az})$ and $(\boldsymbol{X}_{au}, \boldsymbol{Y}_{au})$ are accurate. However, the variables must be linked to the virtual actuator gain $\boldsymbol{M}$ in a consistent way. Therefore, the variables are unified by the substitutions $\boldsymbol{X}_{az} = \boldsymbol{X}_a$ and $\boldsymbol{X}_{au} = \boldsymbol{X}_a$, as well as $\boldsymbol{Y}_{az} = \boldsymbol{Y}_a$ and $\boldsymbol{Y}_{au} = \boldsymbol{Y}_a$, leading to the LMI (8.7), (8.8). The new variables are linked to the gain $\boldsymbol{M}$ by means of the relation $\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1}$, which completes the proof.

## D.5 Proofs of Chapter 10

### *D.5.1 Proof of Theorem 10.1*

Closed-loop stability is established by using Lemma 10.1, the nominal closed-loop IOS (Assumption 9.2), and the established IOS properties of the observation error and difference systems together with Proposition 2.2. The equivalent closed-loop block diagram can be drawn as in Fig. D.2 due to Equation (10.9), (10.10) and interpreted as the cascade interconnection of the nominal closed-loop system with a difference and error subsystem due to Lemma 10.1 (see Remark 10.1).

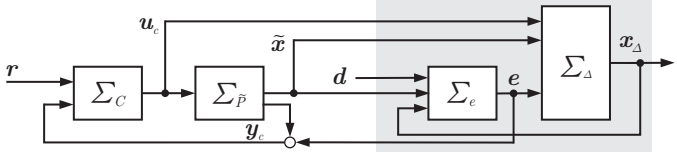


**Fig. D.2** Transformed reconfigured closed-loop system (9.7), (9.6), (10.1), (10.2).

First, consider the feedback connection $(\Sigma_e, \Sigma_\Delta)$ (shaded block in Fig. D.2), to which the signals $\boldsymbol{u}_c$, $\boldsymbol{d}$, and $\tilde{\boldsymbol{x}}$ are external inputs. It was shown in Lemma 10.2 that

$$\|\boldsymbol{e}(t)\| \leq \beta_e(\|\boldsymbol{e}(s)\|, t-s) + \gamma_d(\sup_{s \leq \tau \leq t} \|\boldsymbol{d}(\tau)\|).$$

In other words, the ISS gain of the system $\Sigma_e$ from $\boldsymbol{x}_\Delta$ to $\boldsymbol{e}$ is zero. From the ISS small-gain theorem [97, Theorem 5.6] it follows that the feedback interconnection $(\Sigma_e, \Sigma_\Delta)$ is ISS w.r.t. the input $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}}, \boldsymbol{d})$, hence also IOS w.r.t. the outputs $(\boldsymbol{e}, \boldsymbol{x}_\Delta)$. The ISS small-gain property is now explicitly proven for this particular case. From Lemma 10.2, the observation error dynamics is 0-GES for arbitrary inputs $\tilde{\boldsymbol{x}}$ and $\boldsymbol{x}_\Delta$ and ISS w.r.t. the disturbance $\boldsymbol{d}$. In other words, for $t \geq s$ it is true that

$$\|\boldsymbol{e}(t)\| \leq \beta_e(\|\boldsymbol{e}(s)\|, t-s) + \gamma_d(\sup_{s\leq\tau\leq t} \|\boldsymbol{d}(\tau)\|) \tag{D.25}$$

for some $\beta_e \in \mathcal{KL}$, $\gamma_d \in \mathcal{K}$. The difference system is globally exponentially stable with $\boldsymbol{u}_c = \boldsymbol{e} = \boldsymbol{0}$ for arbitrary $\tilde{\boldsymbol{x}}$ and ISS w.r.t. $\boldsymbol{u}_c$ and $\boldsymbol{e}$, hence it is true for $t \geq s \geq t_0 \geq 0$ that

$$\|\boldsymbol{x}_\Delta(t)\| \leq \beta_\Delta(\|\boldsymbol{x}_\Delta(s)\|, t-s) + \gamma_u\left(\sup_{s\leq\tau\leq t} \|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{s\leq\tau\leq t} \|\boldsymbol{e}(\tau)\|\right) \tag{D.26}$$

for some $\beta_\Delta \in \mathcal{KL}$, $\gamma_u, \gamma_e \in \mathcal{K}$. Applying (D.26) with $s = (t+t_0)/2$ gives

$$\|\boldsymbol{x}_\Delta(t)\| \leq \beta_\Delta\left(\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\|, \frac{t-t_0}{2}\right) + \gamma_u\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{e}(\tau)\|\right). \tag{D.27}$$

To estimate the term $\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\|$, apply (D.26) with $s = t_0$ and $t$ replaced by $(t+t_0)/2$ to get

$$\left\|\boldsymbol{x}_\Delta\left(\frac{t+t_0}{2}\right)\right\| \leq \beta_\Delta\left(\|\boldsymbol{x}_\Delta(t_0)\|, \frac{t-t_0}{2}\right) + \gamma_u\left(\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{e}(\tau)\|\right). \tag{D.28}$$

To estimate the term $\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{e}(\tau)\|$, use (D.25) to obtain

$$\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{e}(\tau)\| \leq \beta_e(\|\boldsymbol{e}(t_0)\|, 0) + \gamma_d\left(\sup_{t_0\leq\tau\leq\frac{t+t_0}{2}} \|\boldsymbol{d}(\tau)\|\right) \tag{D.29}$$

$$\begin{aligned}\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{e}(\tau)\| &\leq \beta_e\left(\left\|\boldsymbol{e}\left(\frac{t+t_0}{2}\right)\right\|, \frac{t-t_0}{2}\right) + \gamma_d\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{d}(\tau)\|\right) \\ &\leq \beta_e\left(\|\boldsymbol{e}(t_0)\|, \frac{t-t_0}{2}\right) + \gamma_d\left(\sup_{\frac{t+t_0}{2}\leq\tau\leq t} \|\boldsymbol{d}(\tau)\|\right).\end{aligned} \tag{D.30}$$

By inserting (D.28)–(D.30) into (D.27) and using the fact that for $\gamma \in \mathcal{K}$ the relation $\gamma(r_1 + r_2) \leq \gamma(2r_1) + \gamma(2r_2)$, the inequality

$$\begin{aligned}
\|\boldsymbol{x}_{\Delta}(t)\| \leq\ & \beta_{\Delta}\left(2\beta_{\Delta}\left(\|\boldsymbol{x}_{\Delta}(t_0)\|, \frac{t-t_0}{2}\right), \frac{t-t_0}{2}\right) \\
& + \beta_{\Delta}\left(2\gamma_e\left(2\beta_e\left(\|\boldsymbol{e}(t_0)\|, 0\right)\right), \frac{t-t_0}{2}\right) + \gamma_e\left(2\beta_e\left(\|\boldsymbol{e}(t_0)\|, \frac{t-t_0}{2}\right)\right) \\
& + \beta_{\Delta}\left(2\gamma_u\left(\sup_{\frac{t+t_0}{2} \leq \tau \leq t} \|\boldsymbol{u}_c(\tau)\|\right), \frac{t-t_0}{2}\right) + \beta_{\Delta}\left(2\gamma_e\left(2\gamma_d\left(\sup_{t_0 \leq \tau \leq \frac{t+t_0}{2}} \|\boldsymbol{d}(\tau)\|\right)\right), \frac{t-t_0}{2}\right) \\
& + \gamma_u\left(\sup_{t_0 \leq \tau \leq \frac{t+t_0}{2}} \|\boldsymbol{u}_c(\tau)\|\right) + \gamma_e\left(2\gamma_d\left(\sup_{\frac{t+t_0}{2} \leq \tau \leq t} \|\boldsymbol{d}(\tau)\|\right)\right)
\end{aligned} \tag{D.31}$$

follows. Clearly, the sum of the first five terms is in class $\mathcal{KL}$, whereas the sum of the last two terms is in class $\mathcal{K}$. Therefore the following shorter notation is introduced,

$$\|\boldsymbol{x}_{\Delta}(t)\| \leq \beta_1(\|\boldsymbol{x}_{\Delta}(t_0)\|, t-t_0) + \beta_2(\|\boldsymbol{e}(t_0)\|, t-t_0) + \gamma_1(\|\boldsymbol{u}_c\|_{\mathcal{L}_\infty}) + \gamma_2(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}), \tag{D.32}$$

where $\beta_1(\cdot) = \beta_{\Delta} \circ 2\beta_{\Delta}(\cdot) \in \mathcal{KL}$ and $\beta_2(\cdot) = \beta_{\Delta} \circ 2\gamma_e \circ 2\beta_e(\cdot) + \gamma_e \circ 2\beta_e(\cdot) \in \mathcal{KL}$ as well as $\gamma_1(\cdot) = \gamma_u(\cdot) + \beta_{\Delta} \circ 2\gamma_u(\cdot) \in \mathcal{K}$ and $\gamma_2(\cdot) = \gamma_e \circ 2\gamma_d(\cdot) + \beta_{\Delta} \circ 2\gamma_e \circ 2\gamma_d(\cdot) \in \mathcal{K}$.

The interconnection $(\Sigma_e, \Sigma_{\Delta})$ satisfies the following relation, obtained by observing that $\|(\boldsymbol{e}^T, \boldsymbol{x}_{\Delta}^T)^T\| \leq \|\boldsymbol{e}\| + \|\boldsymbol{x}_{\Delta}\|$ and $\|(\boldsymbol{u}_c^T, \boldsymbol{d}^T)^T\| \leq \|\boldsymbol{u}_c\| + \|\boldsymbol{d}\|$ and inserting the above relations:

$$\begin{aligned}
\left\|\begin{pmatrix} \boldsymbol{e}(t) \\ \boldsymbol{x}_{\Delta}(t) \end{pmatrix}\right\| \leq\ & \beta_e(\|\boldsymbol{e}(t_0)\|, t-t_0) + \gamma_d(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}) + \beta_1(\|\boldsymbol{x}_{\Delta}(t_0)\|, t-t_0) + \beta_2(\|\boldsymbol{e}(t_0)\|, t-t_0) \\
& + \gamma_1(\|\boldsymbol{u}_c\|_{\mathcal{L}_\infty}) + \gamma_2(\|\boldsymbol{d}\|_{\mathcal{L}_\infty}) \\
\leq\ & \beta_e\left(\left\|\begin{pmatrix} \boldsymbol{x}_{\Delta}(t_0) \\ \boldsymbol{e}(t_0) \end{pmatrix}\right\|, t-t_0\right) + \beta_1\left(\left\|\begin{pmatrix} \boldsymbol{x}_{\Delta}(t_0) \\ \boldsymbol{e}(t_0) \end{pmatrix}\right\|, t-t_0\right) + \beta_2\left(\left\|\begin{pmatrix} \boldsymbol{x}_{\Delta}(t_0) \\ \boldsymbol{e}(t_0) \end{pmatrix}\right\|, t-t_0\right) \\
& + \gamma_d\left(\left\|\begin{pmatrix} \boldsymbol{u}_c \\ \boldsymbol{d} \end{pmatrix}\right\|_{\mathcal{L}_\infty}\right) + \gamma_1\left(\left\|\begin{pmatrix} \boldsymbol{u}_c \\ \boldsymbol{d} \end{pmatrix}\right\|_{\mathcal{L}_\infty}\right) + \gamma_2\left(\left\|\begin{pmatrix} \boldsymbol{u}_c \\ \boldsymbol{d} \end{pmatrix}\right\|_{\mathcal{L}_\infty}\right).
\end{aligned}$$

An ISS-characterisation for the interconnected system $(\Sigma_e, \Sigma_{\Delta})$ has been obtained, where clearly $[\beta_e(r,s) + \beta_1(r,s) + \beta_2(r,s)] \in \mathcal{KL}$ and $[\gamma_d(r) + \gamma_1(r) + \gamma_2(r)] \in \mathcal{K}$ hold. It is concluded that the subsystem $(\Sigma_e, \Sigma_{\Delta})$ is ISS w.r.t. the input $(\boldsymbol{u}_c, \boldsymbol{d}, \tilde{\boldsymbol{x}})$, where, in fact, the ISS gain from $\tilde{\boldsymbol{x}}$ and $\boldsymbol{u}_c$ to $\boldsymbol{e}$ is zero.

The system $(\Sigma_{\tilde{P}}, \Sigma_C)$ is IOS w.r.t. the input $\boldsymbol{r}$ and the output $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}})$ by Assumption 9.2. Using Proposition 2.2 and ignoring the feedback of $\boldsymbol{e}$ to $\boldsymbol{y}_c$, it may be concluded that the series interconnection $((\Sigma_{\tilde{P}}, \Sigma_C), (\Sigma_e, \Sigma_{\Delta}))$ that represents the reconfigured closed-loop system is IOS. Re-introducing the feedback of $\boldsymbol{e}$ maintains the stability property by the ISS small-gain theorem, since the ISS gain from $\boldsymbol{u}_c$ and $\tilde{\boldsymbol{x}}$ to $\boldsymbol{e}$ is zero, which completes the proof.

### D.5.2 Proof of Theorem 10.2

To show ISpS of the error dynamics (10.17) w.r.t. the disturbance input $\boldsymbol{d}$, an ISS-Lyapunov function $V(\boldsymbol{e}) = \frac{1}{2}\boldsymbol{e}^T\boldsymbol{X}\boldsymbol{e}$ is constructed. By using Theorem 9.1 while observing that the error dynamics (10.17) is continuous, one directly obtains for some $a > 0$, $b > 0$, $\theta \in (0,1)$ that

$$\begin{aligned}
\dot{V}(\boldsymbol{e}) = \boldsymbol{e}^T\boldsymbol{X}\dot{\boldsymbol{e}} &= \boldsymbol{e}^T\boldsymbol{X}\left(\boldsymbol{k}_e(\boldsymbol{x}_f+\boldsymbol{e}) - \boldsymbol{k}_e(\boldsymbol{x}_f) - \begin{pmatrix}\boldsymbol{B}_d & \boldsymbol{I}\end{pmatrix}\begin{pmatrix}\boldsymbol{d}\\ \boldsymbol{\varepsilon}(\boldsymbol{x}_f)\end{pmatrix}\right)\\
&\leq -a\boldsymbol{e}^T\boldsymbol{X}\boldsymbol{e} - \boldsymbol{e}^T\boldsymbol{X}\begin{pmatrix}\boldsymbol{B}_d & \boldsymbol{I}\end{pmatrix}\begin{pmatrix}\boldsymbol{d}\\ \boldsymbol{\varepsilon}(\boldsymbol{x}_f)\end{pmatrix}\\
&\leq -b\|\boldsymbol{e}\|^2 + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\boldsymbol{B}_d\|\cdot\|\boldsymbol{d}\| + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\|\cdot\|\boldsymbol{\varepsilon}(\boldsymbol{x}_f)\|\\
&\leq -b\|\boldsymbol{e}\|^2 + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\boldsymbol{B}_d\|\cdot\|\boldsymbol{d}\| + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\|\cdot E\\
&= -(1-\theta)b\|\boldsymbol{e}\|^2 - \theta b\|\boldsymbol{e}\|^2 + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\boldsymbol{B}_d\|\cdot\|\boldsymbol{d}\| + \|\boldsymbol{e}\|\cdot\|\boldsymbol{X}\|\cdot E\\
&\leq -(1-\theta)b\|\boldsymbol{e}\|^2 \text{ if } \|\boldsymbol{e}\| \geq \frac{\|\boldsymbol{X}\|\cdot\|\boldsymbol{B}_d\|}{\theta b}\left(\|\boldsymbol{d}\| + \frac{E}{\|\boldsymbol{B}_d\|}\right),
\end{aligned}$$

which is a Lyapunov characterisation of the ISpS property [89]. In other words, in the absence of disturbances, the observation error converges to a ball with the radius $K = (E\|\boldsymbol{X}\|)/(\theta b)$.

With this result for the ISpS of the observation error, the remaining proof of closed-loop ISpS follows closely along the lines of the proof of Theorem D.5.1, which is not repeated here.

### D.5.3 Proof of Theorem 10.3

Consider the LMIs (10.11) for characterising stabilising gains of the virtual sensor. The substitutions $\boldsymbol{C}_f \to \boldsymbol{B}_f^T$ and $\boldsymbol{A}_i \to \boldsymbol{A}_i^T$ transform the LMIs (10.11) to the LMIs

$$\boldsymbol{X}_s\boldsymbol{A}_i^T + \boldsymbol{A}_i\boldsymbol{X}_s - \boldsymbol{Y}_s\boldsymbol{B}_f^T - \boldsymbol{B}_f\boldsymbol{Y}_s^T \prec 0.$$

By relabeling $\boldsymbol{X}_a \triangleq \boldsymbol{X}_s$ and $\boldsymbol{Y}_a \triangleq \boldsymbol{Y}_s^T$, the LMIs (10.13) have been derived. This result implies that any solution $(\boldsymbol{X}_s, \boldsymbol{Y}_s)$ to the LMIs (10.11) with the parameters $(\boldsymbol{A}_i, \boldsymbol{C}_f)$ corresponding to a PWA system with sensor faults also provides a solution $(\boldsymbol{X}_a, \boldsymbol{Y}_a)$ to the LMIs (10.13) with the parameters $(\boldsymbol{A}_i, \boldsymbol{B}_f)$, which corresponds to a PWA system with actuator faults. The solutions $\boldsymbol{L}$ and $\boldsymbol{M}$ are linked by the relation

$$\boldsymbol{M} = \boldsymbol{Y}_a\boldsymbol{X}_a^{-1} = \boldsymbol{Y}_s^T\boldsymbol{X}_s^{-1} = (\boldsymbol{X}_s^{-1}\boldsymbol{Y}_s)^T = \boldsymbol{L}^T,$$

which completes the proof.

## D.6 Proofs of Chapter 11

### *D.6.1 Proof of Theorem 11.1*

The interconnection $(\bar{\Sigma}_e, \bar{\Sigma}_\Delta)$ is proven to be ISS with respect to the input $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}}, \boldsymbol{d})$ using Lemmas 11.4 and 11.5 as follows. It was shown in Lemma 11.4 that $\|\bar{\boldsymbol{e}}(t)\| \leq ce^{-at}\|\bar{\boldsymbol{e}}(0)\|$ for $t \geq 0$. In other words, the ISS gain of the system $\bar{\Sigma}_e$ from $(\boldsymbol{x}_\Delta, \tilde{\boldsymbol{x}})$ to $\bar{\boldsymbol{e}}$ is zero. The system $\bar{\Sigma}_\Delta$ has finite ISS gain from its inputs $\boldsymbol{u}_c$, $\tilde{\boldsymbol{x}}$, and $\boldsymbol{e}$ to $(\boldsymbol{x}_\Delta, \boldsymbol{x}_I)$. Moreover, the IOS gain (see [89]) from $\tilde{\boldsymbol{x}}$ to $\boldsymbol{x}_\Delta$ is zero. From the ISS small-gain theorem [97, Theorem 5.6] and the IOS small-gain theorem [89, Theorem 2.1], it follows that the feedback interconnection $(\bar{\Sigma}_e, \bar{\Sigma}_\Delta)$ is ISS w.r.t. the input $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}}, \boldsymbol{d})$, hence also IOS w.r.t. the outputs $(\boldsymbol{e}, \boldsymbol{x}_\Delta)$. An explicit proof in terms of this result is straightforward to obtain based on elementary manipulations of the corresponding comparison functions.

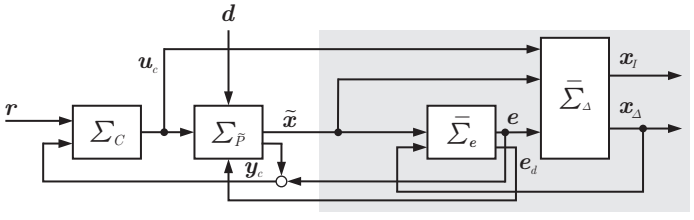


**Fig. D.3** Transformed extended reconfigured closed-loop system (3.5), (9.7), (11.5), (11.11).

Next, the ISS property for the reconfigured extended closed-loop system $(\Sigma_{Pf}, \bar{\Sigma}_S, \bar{\Sigma}_A, \Sigma_C)$ must be verified, which is graphically shown in Fig. D.3. In particular, the feedback signal $\bar{\boldsymbol{e}} = (\boldsymbol{e}^T \ \boldsymbol{e}_d^T)^T$ exponentially converges to zero by Lemma 11.4. The state variable $\boldsymbol{x}_I$ in $\bar{\Sigma}_\Delta$ is not part of a feedback interconnection. However, the state estimation error $\boldsymbol{e}$ and the disturbance observation error $\boldsymbol{e}_d$ are in feedback interconnection with $\bar{\Sigma}_{\tilde{P}}$. Note that by Lemma 11.4, the signal $\boldsymbol{e}_d$ exponentially converges to zero for arbitrary inputs $\boldsymbol{u}_c$, $\tilde{\boldsymbol{x}}$. Therefore, the IOS gain of the system $(\bar{\Sigma}_e, \bar{\Sigma}_\Delta)$ from the input $(\boldsymbol{u}_c, \tilde{\boldsymbol{x}})$ to the output $(\boldsymbol{e}, \boldsymbol{e}_d)$ is zero and it follows from the IOS small-gain theorem [89, Theorem 2.1] and from Assumption 9.3 that the reconfigured extended closed-loop system is ISS. Note that Assumption 9.3 is applicable since $\lim_{t\to\infty} \bar{\boldsymbol{e}}(t) = \boldsymbol{0}$ holds and $(\Sigma_{\tilde{P}}, \Sigma_C)$ is assumed to be ISS w.r.t. $\bar{\boldsymbol{e}}$. Therefore, it has been shown that Problem 9.2 is solved with respect to stability recovery.

It remains to be verified that Problem 9.2 is also solved with respect to set-point tracking recovery. In Chapter 11.3, it has been shown that the reconfigured closed-loop system tracks constant setpoints to precision $K$ provided that the nominal closed-loop system tracks them to this precision, and provided that the extended observation error $\bar{\boldsymbol{e}}$ vanishes and the difference systems state $\boldsymbol{x}_\Delta$ seen through the output matrix $\boldsymbol{C}_z$ vanishes. According to Fig. D.3, the system $(\Sigma_{\tilde{P}}, \Sigma_C)$ is governed by nominal dynamics except for the observation errors $\boldsymbol{e}$ and $\boldsymbol{e}_d$ that perturb the

nominal closed-loop system in the form of intermittent measurement noise, since $\lim_{t\to\infty} \boldsymbol{e}(t) = \boldsymbol{0}$ and $\lim_{t\to\infty} \boldsymbol{e}_d(t) = \boldsymbol{0}$. Consequently, Assumption 9.3 also applies to the system $(\Sigma_{\tilde{P}}, \Sigma_C)$, and the complete solution to Problem 9.2 is provided.

## *D.6.2 Proof of Theorem 11.2*

Noting that the function $\bar{\boldsymbol{k}}_e$ in (11.9) is continuous, a Lyapunov function $V(\bar{\boldsymbol{e}}) = \frac{1}{2}\bar{\boldsymbol{e}}^T \boldsymbol{X}\bar{\boldsymbol{e}}$ is constructed for the system (11.31). The satisfaction of the LMIs (11.19) implies according to Theorem 9.1 that there exist $b > 0$ and $\theta \in (0,1)$ such that

$$\dot{V}(\bar{\boldsymbol{e}}) = \bar{\boldsymbol{e}}^T \boldsymbol{X}\dot{\bar{\boldsymbol{e}}} = \bar{\boldsymbol{e}}^T \boldsymbol{X}\left(\bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}} + \bar{\boldsymbol{e}}) - \bar{\boldsymbol{k}}_e(\bar{\boldsymbol{x}}) - \bar{\boldsymbol{\varepsilon}}(\boldsymbol{x}_f) - \bar{\boldsymbol{\varrho}}\right) \tag{D.33}$$

$$\leq -(1-\theta)b\|\bar{\boldsymbol{e}}\|^2 - \theta b\|\bar{\boldsymbol{e}}\|^2 + \|\bar{\boldsymbol{e}}\| \cdot \|\boldsymbol{X}\|E + \|\bar{\boldsymbol{e}}\| \cdot \|\boldsymbol{X}\| \cdot \|\boldsymbol{\varrho}\| \tag{D.34}$$

$$\leq -(1-\theta)b\|\bar{\boldsymbol{e}}\|^2 \text{ if } \|\bar{\boldsymbol{e}}\| > \frac{\|\boldsymbol{X}\|}{\theta b}(E+F). \tag{D.35}$$

which is a Lyapunov characterisation of the ISpS property [89]. In the presence of disturbance variation, the extended observation error converges to a ball proportional in size to the bound on the disturbance variation $\left(\frac{\|\boldsymbol{X}\|}{\theta b}(E+F)\frac{\lambda_{\max}(\boldsymbol{X})}{\lambda_{\min}(\boldsymbol{X})}\right)$.

With this result for the ISpS of the observation error, the remaining proof of closed-loop ISpS follows closely along the lines of the proof of Theorem D.6.1, which is not repeated here. The proof is based on the observation that the model approximation error $\boldsymbol{\varepsilon}$ only affects the observation error, but neither the difference system, nor the ISS small-gain properties of the interconnection $(\Sigma_e, \Sigma_\Delta)$ (Fig. D.4).

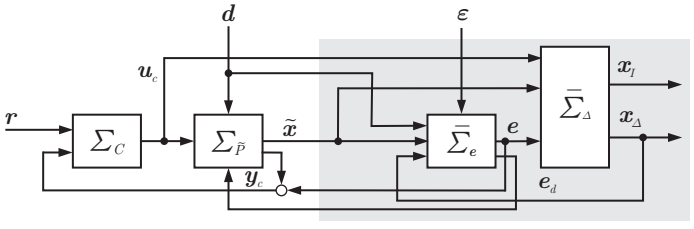


**Fig. D.4** Transformed extended reconfigured closed-loop system (9.6), (9.7), (11.5), (11.11) with modelling error.

The reduced tracking precision follows from the observation that the observation error is bounded by a constant proportional to the model error bound $E$, that the bounded observation error induces a bounded difference system state whose bound is also proportional to the model error bound $E$, and the fact that the steady-state tracking error satisfies the relation $\limsup_{t\to\infty} \|\boldsymbol{e}_z(t)\| = \limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{C}_z\tilde{\boldsymbol{x}}(t) + \boldsymbol{C}_z(\boldsymbol{x}_\Delta(t) + \boldsymbol{e}(t))\|$, where $\limsup_{t\to\infty} \|\boldsymbol{r}(t) - \boldsymbol{C}_z\tilde{\boldsymbol{x}}(t)\| \leq K'$ from Assumption 11.1, $\lim_{t\to\infty} \|\boldsymbol{C}_z\boldsymbol{e}(t)\| \leq c \cdot (E+F)$ for $c = \|\boldsymbol{X}\|/(\theta b)$, and $\lim_{t\to\infty} \|\boldsymbol{C}_z\boldsymbol{x}_\Delta(t)\| \leq c \cdot d \cdot (E+F)$

where $d$ is the ultimate gain of $\bar{\Sigma}_{\Delta}$ w.r.t. the input $\boldsymbol{e}$ and the output $\boldsymbol{x}_{\Delta}$, and therefore $\limsup_{t\to\infty} \|\boldsymbol{e}_z(t)\| \leq K' + c \cdot d \cdot (E + F)$. Note that $\boldsymbol{e}$ and $\boldsymbol{e}_d$ act like persistent measurement disturbances on the system $\Sigma_{\tilde{P}}$.

### D.6.3 Proof of Theorem 11.3

Consider the extended observation error (11.33) and the inputs $\boldsymbol{i}_1(t)$, $\boldsymbol{i}_2(t)$. Denote the ISS gain of the feedback interconnection $(\Sigma_C, \Sigma_{\tilde{P}}, \bar{\Sigma}_e, \bar{\Sigma}_{\Delta})$ from the input $(\boldsymbol{i}_1^T, \boldsymbol{i}_2^T, \boldsymbol{i}_3^T, \boldsymbol{i}_4^T)^T$ to the output $(\tilde{\boldsymbol{x}}^T - \boldsymbol{x}_{\Delta}^T - \boldsymbol{e}^T, \boldsymbol{x}_I^T, \boldsymbol{x}_{\Delta}^T)^T$ by $g$ (this gain exists and is finite, since the new inputs $\boldsymbol{i}_1$ and $\boldsymbol{i}_2$ to $\bar{\Sigma}_e$ enter additively parallel to $\bar{\boldsymbol{\varrho}}$, $\boldsymbol{i}_3$ enters the measurement equation additively parallel to $\boldsymbol{e}$, and $\boldsymbol{i}_4$ enters $\bar{\Sigma}_{\Delta}$ in parallel to $\boldsymbol{u}_c$ and $\boldsymbol{e}$). According to the ISS small-gain theorem, the feedback loop closed by attaching the system with the input $(\tilde{\boldsymbol{x}}^T - \boldsymbol{x}_{\Delta}^T - \boldsymbol{e}^T, \boldsymbol{x}_I^T, \boldsymbol{x}_{\Delta}^T)^T$ and the output $(\boldsymbol{i}_1^T, \boldsymbol{i}_2^T, \boldsymbol{i}_3^T, \boldsymbol{i}_4^T)^T$ is ISS w.r.t. the external input $(\boldsymbol{r}, \boldsymbol{\varrho})$ if Condition (11.36) is satisfied.

### D.6.4 Proof of Theorem 11.4

Consider the LMIs (11.19) for characterising stabilising gains of the virtual sensor. The substitutions $\bar{\boldsymbol{C}}_f \to \bar{\boldsymbol{B}}_f^T$ and $\bar{\boldsymbol{A}}_i \to \bar{\boldsymbol{A}}_i^T$ (which correspond to the substitutions $\boldsymbol{A}_i \to \boldsymbol{A}_i^T$, $\boldsymbol{C}_z \to \boldsymbol{B}_d^T$, and $\boldsymbol{C}_f \to \boldsymbol{B}_f^T$ of the block components) transform the LMIs (11.19) to the LMIs

$$\bar{\boldsymbol{X}}_s \bar{\boldsymbol{A}}_i^T + \bar{\boldsymbol{A}}_i \bar{\boldsymbol{X}}_s - \bar{\boldsymbol{Y}}_s \bar{\boldsymbol{B}}_f^T - \bar{\boldsymbol{B}}_f \bar{\boldsymbol{Y}}_s^T \prec 0.$$

By relabeling $\bar{\boldsymbol{X}}_a \triangleq \bar{\boldsymbol{X}}_s$ and $\bar{\boldsymbol{Y}}_a \triangleq \bar{\boldsymbol{Y}}_s^T$, the LMI (11.23) has been derived. This result implies that any solution $(\bar{\boldsymbol{X}}_s, \bar{\boldsymbol{Y}}_s)$ to the LMIs (11.19) with the parameters $(\bar{\boldsymbol{C}}_f, \bar{\boldsymbol{A}}_i)$ corresponding to a PWA system with sensor faults also provides a solution $(\bar{\boldsymbol{X}}_a, \bar{\boldsymbol{Y}}_a)$ to the LMIs (11.23) for the system with the parameters $(\bar{\boldsymbol{A}}_i^T, \bar{\boldsymbol{C}}_f^T)$, which corresponds to a PWA system with actuator faults. The solutions $\bar{\boldsymbol{L}}$ and $\bar{\boldsymbol{M}}$ are linked by the relation

$$\bar{\boldsymbol{M}} = \bar{\boldsymbol{Y}}_a \bar{\boldsymbol{X}}_a^{-1} = \bar{\boldsymbol{Y}}_s^T \bar{\boldsymbol{X}}_s^{-1} = (\bar{\boldsymbol{X}}_s^{-1} \bar{\boldsymbol{Y}}_s)^T = \bar{\boldsymbol{L}}^T,$$

which completes the proof.

# Appendix E
# Models of the Thermofluid Process

## E.1 Nonlinear Process Model

Using mass and enthalpy balances [81], the nonlinear state space model for the thermofluid process is obtained as follows. The dependence of quantities on time is not denoted explicitly for the sake of simplicity. The state equation has the structure (3.2) with

$$
\begin{aligned}
f_1(\boldsymbol{x},\boldsymbol{u}) &= \frac{1}{A\rho l_{\mathrm{TS}}}\left(\frac{P_{\mathrm{el,TS}}(\boldsymbol{u})-\dot{Q}_{\mathrm{V,TS}}(\boldsymbol{x},\boldsymbol{u})}{\mathrm{c_p}}+k_1(\boldsymbol{x},\boldsymbol{u})\right),\\
f_2(\boldsymbol{x},\boldsymbol{u}) &= \frac{1}{A\rho}\left(\dot{m}_{\mathrm{TB}}(\boldsymbol{u})+\dot{m}_{\mathrm{TM}}(\boldsymbol{u})+\dot{m}_{\mathrm{CW}}(\boldsymbol{u})-\dot{m}_{\mathrm{TW}}(\boldsymbol{x},\boldsymbol{u})\right),\\
f_3(\boldsymbol{x},\boldsymbol{u}) &= \frac{1}{A\rho l_{\mathrm{TS}}}\left(\dot{m}_{\mathrm{TB}}(\boldsymbol{u})(c_{\mathrm{TB}}-c_{\mathrm{TS}})+\dot{m}_{\mathrm{TM}}(\boldsymbol{u})(c_{\mathrm{TM}}-c_{\mathrm{TS}})-\dot{m}_{\mathrm{CW}}(\boldsymbol{u})c_{\mathrm{TS}}\right),\\
f_4(\boldsymbol{x},\boldsymbol{u}) &= \frac{1}{A\rho l_{\mathrm{TB}}}\left(\frac{P_{\mathrm{el,TB}}(\boldsymbol{u})-\dot{Q}_{\mathrm{V,TB}}(\boldsymbol{x},\boldsymbol{u})}{\mathrm{c_p}}+\dot{m}_{\mathrm{TV}}(\vartheta_{\mathrm{TV}}-\vartheta_{\mathrm{TB}})\right),\\
f_5(\boldsymbol{x},\boldsymbol{u}) &= -\frac{1}{T_{\mathrm{CW}}}x_{\mathrm{CW}}+\frac{1}{T_{\mathrm{CW}}}u_{\mathrm{CW}},\\
f_6(\boldsymbol{x},\boldsymbol{u}) &= -\frac{1}{T_{\mathrm{el,TB}}}x_{\mathrm{el,TB}}+\frac{1}{T_{\mathrm{el,TB}}}u_{\mathrm{el,TB}},\\
f_7(\boldsymbol{x},\boldsymbol{u}) &= -\frac{1}{T_{\mathrm{el,TS}}}x_{\mathrm{el,TS}}+\frac{1}{T_{\mathrm{el,TS}}}u_{\mathrm{el,TS}},
\end{aligned}
$$

and

$$
k_1(\boldsymbol{x},\boldsymbol{u}) = \dot{m}_{\mathrm{TB}}(\boldsymbol{u})(\vartheta_{\mathrm{TB}}-\vartheta_{\mathrm{TS}})+\dot{m}_{\mathrm{TM}}(\boldsymbol{u})(\vartheta_{\mathrm{TM}}-\vartheta_{\mathrm{TS}})+\dot{m}_{\mathrm{CW}}(\boldsymbol{u})(\vartheta_{\mathrm{CW}}-\vartheta_{\mathrm{TS}}).
$$

The output equation of the model (3.2) has the entries

$$
\begin{aligned}
h_1(\boldsymbol{x}) &= \vartheta_{\mathrm{TS}},\\
h_2(\boldsymbol{x}) &= l_{\mathrm{TS}},
\end{aligned}
$$

$$h_3(\boldsymbol{x}) = 0.4469\,\frac{\mathrm{mS}}{\mathrm{cm}} + 2047.7\,\frac{\mathrm{mS}}{\mathrm{cm}}c_{\mathrm{TS}},$$
$$h_4(\boldsymbol{x}) = \vartheta_{\mathrm{TB}}.$$

The parameters of the nonlinear model are given with their values in Table E.1. Below, each term is presented in more detail by plant component.

**Table E.1** Nonlinear state space model parameters.

| Parameter | Value |
|---|---|
| $T_{\mathrm{CW}}$ | 3.7 s |
| $T_{\mathrm{el,TB}}$ | 27 s |
| $T_{\mathrm{el,TS}}$ | 65 s |
| $c_{\mathrm{TM}}$ | $1.97930 \cdot 10^{-3}$ |
| $c_{\mathrm{TB}}$ | $1.73513 \cdot 10^{-3}$ |
| $l_{\mathrm{TB}}$ | 0.41 m |
| $l_{\mathrm{TM}}$ | 0.31 m |
| $\vartheta_{\mathrm{CW}}$ | 25 °C |
| $A$ | 0.707 $\mathrm{m}^2$ |
| $\vartheta_{\mathrm{U}}$ | 22.5 °C |

**Cold water supply $u_{\mathrm{CW}}$.** The subordinate control of the cold water mass flow rate $\dot{m}_{\mathrm{CW}}$ into the tank TS is modelled as a controlled actuator, with the desired mass flow rate as its input, and is modelled as a first order delay block

$$\dot{x}_{\mathrm{CW}}(t) = -\frac{1}{T_{\mathrm{CW}}}x_{\mathrm{CW}}(t) + \frac{1}{T_{\mathrm{CW}}}u_{\mathrm{CW}}(t),$$
$$\dot{m}_{\mathrm{CW}}(t) = x_{\mathrm{CW}}(t).$$

**Pump $u_{\mathrm{PS}}$.** The mass flow rate $\dot{m}_{\mathrm{TW}}$ from the tank TS is controlled by means of the pump. Since the smallest possible mass flow is too large for the considered process, pulse-width modulation (PWM) is used to realise smaller flow rates:

$$u_{\mathrm{PS}}(t) = \frac{T_{\mathrm{on}}(t)}{T_{\mathrm{PWM}}},$$

with a period $T_{\mathrm{PWM}} = 6$ s. Taking the elevation of 0.36 m into account, the mass flow rate out of the tank TS into the tank TW is

$$\dot{m}_{\mathrm{TW}}(t) = 0.1679\,\frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}}u_{\mathrm{PS}}(t)\sqrt{l_{\mathrm{TS}}(t) + 0.36\ \mathrm{m}}.$$

**Control valve $u_{\mathrm{TB}}$.** The control of the mass flow rate $\dot{m}_{\mathrm{TB}}$ from TB into TS uses the control valve $u_{\mathrm{TB}}$. The mass flow through the valve is subject to a dead zone due to combined pump and valve properties as

$$\dot{m}_{\mathrm{TB}}(t) = \begin{cases} (c_1 + c_2(u_{\mathrm{TB}}(t) - D_1)) \cdot \sqrt{l_{\mathrm{TB}} + 0.3\ \mathrm{m}}, & \text{if } u_{\mathrm{TB}} \geq D_1 \\ 0\ \frac{\mathrm{kg}}{\mathrm{s}}, & \text{if } u_{\mathrm{TB}} < D_1 \end{cases}$$

with the parameters $c_1 = 0.019\ \mathrm{kg/(s\sqrt{m})}$, $c_2 = 0.727\ \mathrm{kg/(s\sqrt{m})}$ and a dead zone $D_1 = 0.13$.

**Control valve $u_{\mathrm{TM}}$.** The control of the mass flow $\dot{m}_{\mathrm{TM}}$ from TM into TS happens analogously to $\dot{m}_{\mathrm{TB}}$ using the control valve $u_{\mathrm{TM}}$, yielding

$$\dot{m}_{\mathrm{TM}}(t) = \begin{cases} (c_3 + c_4(u_{\mathrm{TM}}(t) - D_2)) \cdot \sqrt{l_{\mathrm{TM}} + 0.3\ \mathrm{m}}, & \text{if } u_{\mathrm{TM}} \geq D_2 \\ 0\ \frac{\mathrm{kg}}{\mathrm{s}}, & \text{if } u_{\mathrm{TM}} < D_2, \end{cases} \tag{E.1}$$

with the parameters $c_3 = 0.047\ \mathrm{kg/(s\sqrt{m})}$, $c_4 = 0.605\ \mathrm{kg/(s\sqrt{m})}$ and a dead zone $D_2 = 0.04$.

**Heater $u_{\mathrm{el,TS}}$.** The heater $u_{\mathrm{el,TS}}$ in the reactor TS consist of 4 heating rods, each with 1 kW power. Every heating rod can only be turned completely on or off, hence PWM is used to provide a pseudo-continuous power range. The heater $u_{\mathrm{el,TS}}$ is modelled as a first order delay block with input $u_{\mathrm{el,TS}}$ and output $P_{\mathrm{el,TS}}$

$$\begin{aligned} \dot{x}_{\mathrm{el,TS}}(t) &= -\frac{1}{T_{\mathrm{el,TS}}} x_{\mathrm{el,TS}}(t) + \frac{1}{T_{\mathrm{el,TS}}} u_{\mathrm{el,TS}}(t), \\ P_{\mathrm{el,TS}}(t) &= k_{\mathrm{el,TS}} x_{\mathrm{el,TS}}(t), \end{aligned}$$

where $k_{\mathrm{el,TS}} = 4$ kW.

**Heat loss of the reactor TS.** By experience, the heat loss $\dot{Q}_{\mathrm{V,TS}}$ depends on the activity of the heating rods:

$$\dot{Q}_{\mathrm{V,TS}}(\vartheta_{\mathrm{TS}}(t)) = \begin{cases} \dot{Q}^{\mathrm{on}}_{\mathrm{V,TS}}(\vartheta_{\mathrm{TS}}(t)), & \text{if } u_{\mathrm{el,TS}} > 0 \\ \dot{Q}^{\mathrm{off}}_{\mathrm{V,TS}}(\vartheta_{\mathrm{TS}}(t)), & \text{if } u_{\mathrm{el,TS}} = 0. \end{cases}$$

The individual heat losses are

$$\dot{Q}^{\mathrm{on}}_{\mathrm{V,TS}}(\vartheta_{\mathrm{TS}}(t)) = \begin{cases} c_5(\vartheta_{\mathrm{TS}}(t) - \vartheta_{\mathrm{U}}), & \text{if } \vartheta_{\mathrm{TS}} \geq \vartheta_{\mathrm{U}} \\ 0\mathrm{W}, & \text{if } \vartheta_{\mathrm{TS}} < \vartheta_{\mathrm{U}}, \end{cases}$$

$$\dot{Q}^{\mathrm{off}}_{\mathrm{V,TS}}(\vartheta_{\mathrm{TS}}(t)) = \begin{cases} c_6(\vartheta_{\mathrm{TS}}(t) - \vartheta_{\mathrm{U}}), & \text{if } \vartheta_{\mathrm{TS}} \geq \vartheta_{\mathrm{U}} \\ 0\mathrm{W}, & \text{if } \vartheta_{\mathrm{TS}} < \vartheta_{\mathrm{U}}, \end{cases}$$

where $c_5 = 46.9403$ W/K, $c_6 = 4.8968$ W/K and the ambient temperature $\vartheta_{\mathrm{U}} = 22.5\ ^{\circ}\mathrm{C}$. Heat transfer into TS in case the ambient air is warmer than the reactor content ($\vartheta_{\mathrm{TS}} < \vartheta_{\mathrm{U}}$) is neglected, because the process is only operated above ambient temperature.

**Heater $u_{\mathrm{el,TB}}$.** The heater $u_{\mathrm{el,TB}}$ in the reactor TB consists of 6 heating rods with 3 kW power each. As in TS, the heating rods are operated using PWM. The 6 rods are modelled as a first order delay block with input $u_{\mathrm{el,TB}}$ and output $P_{\mathrm{el,TB}}$,

$$\begin{aligned}\dot{x}_{\mathrm{el,TB}}(t) &= -\frac{1}{T_{\mathrm{el,TB}}}x_{\mathrm{el,TB}}(t) + \frac{1}{T_{\mathrm{el,TB}}}u_{\mathrm{el,TB}}(t),\\ P_{\mathrm{el,TB}}(t) &= k_{\mathrm{el,TB}}x_{\mathrm{el,TB}}(t),\end{aligned}$$

where $k_{\mathrm{el,TB}} = 18$ kW.

**Heat loss of the reactor TB.** The heat loss $\dot{Q}_{\mathrm{V,TB}}$ in TB is modelled depending on the heating rod operation as for TS:

$$\dot{Q}_{\mathrm{V,TB}}(\vartheta_{\mathrm{TB}}(t)) = \begin{cases}\dot{Q}^{\mathrm{on}}_{\mathrm{V,TB}}(\vartheta_{\mathrm{TB}}(t)), & \text{if } u_{\mathrm{el,TB}} > 0\\ \dot{Q}^{\mathrm{off}}_{\mathrm{V,TB}}(\vartheta_{\mathrm{TB}}(t)), & \text{if } u_{\mathrm{el,TB}} = 0,\end{cases}$$

with the loss terms

$$\dot{Q}^{\mathrm{on}}_{\mathrm{V,TB}}(\vartheta_{\mathrm{TB}}(t)) = \begin{cases}c_7(\vartheta_{\mathrm{TB}}(t) - \vartheta_{\mathrm{U}}), & \text{if } \vartheta_{\mathrm{TB}} \geq \vartheta_{\mathrm{U}}\\ 0\mathrm{W}, & \text{if } \vartheta_{\mathrm{TB}} < \vartheta_{\mathrm{U}},\end{cases}$$

$$\dot{Q}^{\mathrm{off}}_{\mathrm{V,TB}}(\vartheta_{\mathrm{TB}}(t)) = \begin{cases}c_8(\vartheta_{\mathrm{TB}}(t) - \vartheta_{\mathrm{U}}), & \text{if } \vartheta_{\mathrm{TB}} \geq \vartheta_{\mathrm{U}}\\ 0\mathrm{W}, & \text{if } \vartheta_{\mathrm{TB}} < \vartheta_{\mathrm{U}},\end{cases}$$

and parameters $c_7 = 135.468$ W/K and $c_8 = 4.8968$ W/K. As for TS, heat transfer into TB is not modelled due to the process operating exclusively above the ambient temperature.

To linearise this model around the equilibrium given in Table E.2 it must be noted that due to the dead zone (Equation (E.1)) the states are decoupled from the input signal $u_{\mathrm{TM}}$. To preserve this dependency, the mass flow rate from TM into TS is modelled for the purpose of linearisation as

$$\dot{m}_{\mathrm{TM}}(t) = 0.6428\ \frac{\mathrm{kg}}{\mathrm{s}\sqrt{\mathrm{m}}}u_{\mathrm{TM}}(t)\sqrt{l_{\mathrm{TM}} + 0.3\ \mathrm{m}}.$$

**Table E.2** Equilibrium point used for operation and linearisation

| Equilibrium value | Equilibrium value |
|---|---|
| $\bar{\vartheta}_{\mathrm{TS}} = 25\ ^\circ\mathrm{C}$ | $\bar{c}_{\mathrm{TS}} = 8.07 \cdot 10^{-4}\%$ |
| $\bar{l}_{\mathrm{TS}} = 0.335$ m | $\bar{u}_{\mathrm{TB}} = 0.1618$ |
| $\bar{\vartheta}_{\mathrm{TB}} = 25\ ^\circ\mathrm{C}$ | $\bar{u}_{\mathrm{PS}} = 0.603$ |
| $\bar{u}_{\mathrm{TM}} = 0$ | $\bar{\dot{x}}_{\mathrm{el,TS}} = \bar{u}_{\mathrm{el,TS}} = 0.5181$ |
| $\bar{u}_{\mathrm{CW}} = \bar{x}_{\mathrm{CW}} = 0.04$ kg/s | $\bar{\dot{x}}_{\mathrm{el,TB}} = \bar{u}_{\mathrm{el,TB}} = 6.93 \cdot 10^{-4}$ |

## E.2 Piecewise Affine Process Model

The following piecewise affine model defined on 48 polytopes is used in this monograph.

$$A_1 = \begin{pmatrix} -0.009471 & -0.03673 & 0 & 0.007377 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.00899 & 4.106e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} -0.009471 & -0.03673 & 0 & 0.007377 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.00899 & 4.106e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} -0.00451 & 0.8653 & 0 & 0.01775 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.004281 & 3.78e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_4 = \begin{pmatrix} -0.009471 & -0.03673 & 0 & 0.007377 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.004281 & 3.78e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_5 = \begin{pmatrix} -0.00451 & 0.8653 & 0 & 0.01775 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.004281 & 3.78e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_6 = \begin{pmatrix} -0.00451 & 0.8653 & 0 & 0.01775 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & -6.574e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.00899 & 4.106e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003265 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_7 = \begin{pmatrix} -0.008255 & -0.03673 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.106e-07 & -2.545e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_8 = \begin{pmatrix} -0.008255 & -0.03673 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.106e-07 & -2.545e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_9 = \begin{pmatrix} -0.008255 & -0.03673 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 3.78e-07 & 1.736e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{10} = \begin{pmatrix} -0.01251 & -0.4768 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 3.78e-07 & 1.736e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{11} = \begin{pmatrix} -0.01251 & -0.4768 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.106e-07 & -2.545e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{12} = \begin{pmatrix} -0.01251 & -0.4768 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ -6.574e-07 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 3.78e-07 & 1.736e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 1.851e-05 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{13} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{14} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{15} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{16} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{17} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{A}_{18} = \begin{pmatrix} -0.00451 & -0.006279 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0007393 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -4.879e-06 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ 0 & -0.001089 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{19} = \begin{pmatrix} -0.00451 & -0.03673 & 2.17 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{20} = \begin{pmatrix} -0.00451 & -0.03673 & 2.17 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{21} = \begin{pmatrix} -0.009471 & 0.8653 & 83.47 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{22} = \begin{pmatrix} -0.009471 & 0.4252 & 47.19 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{23} = \begin{pmatrix} -0.00451 & -0.4768 & -34.11 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{24} = \begin{pmatrix} -0.009471 & 0.4252 & 47.19 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0.002756 & 0 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.005866 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & -0.07761 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{25} = \begin{pmatrix} -0.009471 & -0.03673 & 7.533 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{26} = \begin{pmatrix} -0.009471 & -0.03673 & 7.533 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{27} = \begin{pmatrix} -0.00451 & 0.4252 & 45.43 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{28} = \begin{pmatrix} -0.00451 & 0.8653 & 71.78 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{29} = \begin{pmatrix} -0.009471 & -0.4768 & -18.82 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{30} = \begin{pmatrix} -0.00451 & 0.4252 & 45.43 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & -0.004072 & 0 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.006447 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0.1147 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{31} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{32} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{33} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{34} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{35} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{36} = \begin{pmatrix} -0.009471 & 0.008315 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 0 & -0.0008023 & 0 & 0 & 0.001415 & 0 & 0 \\ 0 & -1.025e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ 0 & 0.0006857 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{37} = \begin{pmatrix} -0.00451 & 0.4252 & 0 & 0.001467 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.004281 & 6.995e-08 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{38} = \begin{pmatrix} -0.009471 & -0.4768 & 0 & 0.003541 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.004281 & 6.995e-08 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{39} = \begin{pmatrix} -0.009471 & -0.4768 & 0 & 0.003541 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.00899 & 1.491e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{40} = \begin{pmatrix} -0.009471 & -0.4768 & 0 & 0.003541 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.00899 & 1.491e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{41} = \begin{pmatrix} -0.00451 & 0.4252 & 0 & 0.001467 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & -2.545e-05 & -0.004281 & 6.995e-08 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{42} = \begin{pmatrix} -0.00451 & 0.4252 & 0 & 0.001467 & -0.05912 & 0 & 0.04042 \\ 0 & -0.000778 & 0 & 1.315e-07 & 0.001415 & 0 & 0 \\ 0 & 1.736e-05 & -0.00899 & 1.491e-07 & -3.409e-06 & 0 & 0 \\ 0 & 0 & 0 & -0.003287 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{43} = \begin{pmatrix} -0.005558 & 0.4252 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.996e-08 & -2.545e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{44} = \begin{pmatrix} -0.006166 & 0.8653 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.996e-08 & -2.545e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{45} = \begin{pmatrix} -0.006166 & 0.8653 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 1.065e-07 & 1.736e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{46} = \begin{pmatrix} -0.006166 & 0.8653 & 0 & 0.006161 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 1.065e-07 & 1.736e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{47} = \begin{pmatrix} -0.005558 & 0.4252 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 4.996e-08 & -2.545e-05 & -0.004281 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$A_{48} = \begin{pmatrix} -0.005558 & 0.4252 & 0 & 0.002934 & -0.05912 & 0 & 0.04042 \\ 9.391e-08 & -0.000778 & 0 & 0 & 0.001415 & 0 & 0 \\ 1.065e-07 & 1.736e-05 & -0.00899 & 0 & -3.409e-06 & 0 & 0 \\ -2.645e-06 & 0 & 0 & -0.003283 & 0 & 0.1604 & 0 \\ 0 & 0 & 0 & 0 & -0.2703 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.03704 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.01538 \end{pmatrix}$$

$$\boldsymbol{a}_1 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_2 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_3 = \begin{pmatrix} -0.621 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_4 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_5 = \begin{pmatrix} -0.621 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_6 = \begin{pmatrix} -0.621 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_7 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_8 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_9 = \begin{pmatrix} 0.06453 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{10} = \begin{pmatrix} 0.399 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{11} = \begin{pmatrix} 0.399 \\ 0.0002771 \\ 5.52e-06 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{12} = \begin{pmatrix} 0.399 \\ 0.0002771 \\ -1.181e-05 \\ 0.08151 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{13} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{14} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{15} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{16} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{17} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{18} = \begin{pmatrix} 0.04139 \\ 0.0002477 \\ 5.091e-06 \\ 0.08234 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{19} = \begin{pmatrix} 0.04983 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{20} = \begin{pmatrix} 0.04983 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{21} = \begin{pmatrix} -0.1939 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{22} = \begin{pmatrix} -0.09791 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{23} = \begin{pmatrix} 0.1459 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{24} = \begin{pmatrix} -0.09791 \\ 0.0002584 \\ -1.08e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{25} = \begin{pmatrix} 0.08885 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{26} = \begin{pmatrix} 0.08885 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{27} = \begin{pmatrix} -0.1398 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{28} = \begin{pmatrix} -0.3892 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{29} = \begin{pmatrix} 0.3382 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{30} = \begin{pmatrix} -0.1398 \\ 0.0002639 \\ 1.373e-05 \\ 0.08188 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{31} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{32} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{33} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{34} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{35} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{36} = \begin{pmatrix} 0.07984 \\ 0.0002688 \\ 1.069e-05 \\ 0.08174 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{37} = \begin{pmatrix} -0.06649 \\ 0.0002573 \\ 1.023e-05 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{38} = \begin{pmatrix} 0.3078 \\ 0.0002573 \\ 1.023e-05 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{39} = \begin{pmatrix} 0.3078 \\ 0.0002573 \\ -2.284e-06 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{40} = \begin{pmatrix} 0.3078 \\ 0.0002573 \\ -2.284e-06 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{41} = \begin{pmatrix} -0.06649 \\ 0.0002573 \\ 1.023e-05 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{42} = \begin{pmatrix} -0.06649 \\ 0.0002573 \\ -2.284e-06 \\ 0.08207 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{43} = \begin{pmatrix} -0.07696 \\ 0.0002583 \\ 1.073e-05 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{44} = \begin{pmatrix} -0.2899 \\ 0.0002583 \\ 1.073e-05 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{a}_{45} = \begin{pmatrix} -0.2899 \\ 0.0002583 \\ -1.22e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{46} = \begin{pmatrix} -0.2899 \\ 0.0002583 \\ -1.22e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{47} = \begin{pmatrix} -0.07696 \\ 0.0002583 \\ 1.073e-05 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{a}_{48} = \begin{pmatrix} -0.07696 \\ 0.0002583 \\ -1.22e-06 \\ 0.08204 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$\boldsymbol{B} = \begin{pmatrix} -0.01062 & 0 & 0 & -0.001062 & 0 & 0 \\ 0.007117 & 0.007616 & 0 & 0 & 0 & -0.001984 \\ 2.49e-05 & 2.109e-05 & 0 & 0 & 0 & 0 \\ 0 & 0 & -0.002908 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.2703 & 0 \\ 0 & 0 & 0.03704 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01538 & 0 & 0 \end{pmatrix}$$

$$\boldsymbol{C} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2048 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

# Index

# Lecture Notes in Control and Information Sciences

**Edited by M. Thoma, F. Allgöwer, M. Morari**

Further volumes of this series can be found on our homepage:
springer.com

**Vol. 408:** Richter, J.H.;
Reconfigurable Control of
Nonlinear Dynamical Systems
291 p. 2011 [978-3-642-17627-2]

**Vol. 407:** Lévine, J., Müllhaupt, P.:
Advances in the Theory of Control,
Signals and Systems with
Physical Modeling
380 p. 2010 [978-3-642-16134-6]

**Vol. 406:** Bemporad, A., Heemels, M.,
Johansson, M.:
Networked Control Systems
appro. 371 p. 2010 [978-0-85729-032-8]

**Vol. 405:** Stefanovic, M., Safonov, M.G.:
Safe Adaptive Control
appro. 153 p. 2010 [978-1-84996-452-4]

**Vol. 404:** Giri, F.; Bai, E.-W. (Eds.):
Block-oriented Nonlinear System Identification
425 p. 2010 [978-1-84996-512-5]

**Vol. 403:** Tóth, R.;
Modeling and Identification of
Linear Parameter-Varying Systems
319 p. 2010 [978-3-642-13811-9]

**Vol. 402:** del Re, L.; Allgöwer, F.;
Glielmo, L.; Guardiola, C.;
Kolmanovsky, I. (Eds.):
Automotive Model Predictive Control
284 p. 2010 [978-1-84996-070-0]

**Vol. 401:** Chesi, G.; Hashimoto, K. (Eds.):
Visual Servoing via Advanced
Numerical Methods
393 p. 2010 [978-1-84996-088-5]

**Vol. 400:** Tomás-Rodríguez, M.;
Banks, S.P.:
Linear, Time-varying Approximations
to Nonlinear Dynamical Systems
298 p. 2010 [978-1-84996-100-4]

**Vol. 399:** Edwards, C.; Lombaerts, T.;
Smaili, H. (Eds.):
Fault Tolerant Flight Control
appro. 350 p. 2010 [978-3-642-11689-6]

**Vol. 398:** Hara, S.; Ohta, Y.;
Willems, J.C.; Hisaya, F. (Eds.):
Perspectives in Mathematical System
Theory, Control, and Signal Processing
appro. 370 p. 2010 [978-3-540-93917-7]

**Vol. 397:** Yang, H.; Jiang, B.;
Cocquempot, V.:
Fault Tolerant Control Design for
Hybrid Systems
191 p. 2010 [978-3-642-10680-4]

**Vol. 396:** Kozlowski, K. (Ed.):
Robot Motion and Control 2009
475 p. 2009 [978-1-84882-984-8]

**Vol. 395:** Talebi, H.A.; Abdollahi, F.;
Patel, R.V.; Khorasani, K.:
Neural Network-Based State
Estimation of Nonlinear Systems
appro. 175 p. 2010 [978-1-4419-1437-8]

**Vol. 394:** Pipeleers, G.; Demeulenaere, B.;
Swevers, J.:
Optimal Linear Controller Design for
Periodic Inputs
177 p. 2009 [978-1-84882-974-9]

**Vol. 393:** Ghosh, B.K.; Martin, C.F.;
Zhou, Y.:
Emergent Problems in Nonlinear
Systems and Control
285 p. 2009 [978-3-642-03626-2]

**Vol. 392:** Bandyopadhyay, B.;
Deepak, F.; Kim, K.-S.:
Sliding Mode Control Using Novel Sliding
Surfaces
137 p. 2009 [978-3-642-03447-3]

**Vol. 391:** Khaki-Sedigh, A.; Moaveni, B.:
Control Configuration Selection for
Multivariable Plants
232 p. 2009 [978-3-642-03192-2]

**Vol. 390:** Chesi, G.; Garulli, A.;
Tesi, A.; Vicino, A.:
Homogeneous Polynomial Forms for
Robustness Analysis of Uncertain
Systems
197 p. 2009 [978-1-84882-780-6]

**Vol. 389:** Bru, R.; Romero-Vivó, S. (Eds.):
Positive Systems
398 p. 2009 [978-3-642-02893-9]

**Vol. 388:** Jacques Loiseau, J.; Michiels, W.; Niculescu, S-I.; Sipahi, R. (Eds.):
Topics in Time Delay Systems
418 p. 2009 [978-3-642-02896-0]

**Vol. 387:** Xia, Y.; Fu, M.; Shi, P.:
Analysis and Synthesis of Dynamical Systems with Time-Delays
283 p. 2009 [978-3-642-02695-9]

**Vol. 386:** Huang, D.; Nguang, S.K.:
Robust Control for Uncertain Networked Control Systems with Random Delays
159 p. 2009 [978-1-84882-677-9]

**Vol. 385:** Jungers, R.:
The Joint Spectral Radius
144 p. 2009 [978-3-540-95979-3]

**Vol. 384:** Magni, L.; Raimondo, D.M.; Allgöwer, F. (Eds.):
Nonlinear Model Predictive Control
572 p. 2009 [978-3-642-01093-4]

**Vol. 383:** Sobhani-Tehrani E.; Khorasani K.;
Fault Diagnosis of Nonlinear Systems Using a Hybrid Approach
360 p. 2009 [978-0-387-92906-4]

**Vol. 382:** Bartoszewicz A.; Nowacka-Leverton A.;
Time-Varying Sliding Modes for Second and Third Order Systems
192 p. 2009 [978-3-540-92216-2]

**Vol. 381:** Hirsch M.J.; Commander C.W.; Pardalos P.M.; Murphey R. (Eds.)
Optimization and Cooperative Control Strategies: Proceedings of the 8th International Conference on Cooperative Control and Optimization
459 p. 2009 [978-3-540-88062-2]

**Vol. 380:** Basin M.
New Trends in Optimal Filtering and Control for Polynomial and Time-Delay Systems
206 p. 2008 [978-3-540-70802-5]

**Vol. 379:** Mellodge P.; Kachroo P.;
Model Abstraction in Dynamical Systems: Application to Mobile Robot Control
116 p. 2008 [978-3-540-70792-9]

**Vol. 378:** Femat R.; Solis-Perales G.;
Robust Synchronization of Chaotic Systems Via Feedback
199 p. 2008 [978-3-540-69306-2]

**Vol. 377:** Patan K.
Artificial Neural Networks for the Modelling and Fault Diagnosis of Technical Processes
206 p. 2008 [978-3-540-79871-2]

**Vol. 376:** Hasegawa Y.
Approximate and Noisy Realization of Discrete-Time Dynamical Systems
245 p. 2008 [978-3-540-79433-2]

**Vol. 375:** Bartolini G.; Fridman L.; Pisano A.; Usai E. (Eds.)
Modern Sliding Mode Control Theory
465 p. 2008 [978-3-540-79015-0]

**Vol. 374:** Huang B.; Kadali R.
Dynamic Modeling, Predictive Control and Performance Monitoring
240 p. 2008 [978-1-84800-232-6]

**Vol. 373:** Wang Q.-G.; Ye Z.; Cai W.-J.; Hang C.-C.
PID Control for Multivariable Processes
264 p. 2008 [978-3-540-78481-4]

**Vol. 372:** Zhou J.; Wen C.
Adaptive Backstepping Control of Uncertain Systems
241 p. 2008 [978-3-540-77806-6]

**Vol. 371:** Blondel V.D.; Boyd S.P.; Kimura H. (Eds.)
Recent Advances in Learning and Control
279 p. 2008 [978-1-84800-154-1]

**Vol. 370:** Lee S.; Suh I.H.; Kim M.S. (Eds.)
Recent Progress in Robotics: Viable Robotic Service to Human
410 p. 2008 [978-3-540-76728-2]

**Vol. 369:** Hirsch M.J.; Pardalos P.M.; Murphey R.; Grundel D.
Advances in Cooperative Control and Optimization
423 p. 2007 [978-3-540-74354-5]

**Vol. 368:** Chee F.; Fernando T.
Closed-Loop Control of Blood Glucose
157 p. 2007 [978-3-540-74030-8]

**Vol. 367:** Turner M.C.; Bates D.G. (Eds.)
Mathematical Methods for Robust and Nonlinear Control
444 p. 2007 [978-1-84800-024-7]

**Vol. 366:** Bullo F.; Fujimoto K. (Eds.)
Lagrangian and Hamiltonian Methods for Nonlinear Control 2006
398 p. 2007 [978-3-540-73889-3]

**Vol. 365:** Bates D.; Hagström M. (Eds.)
Nonlinear Analysis and Synthesis Techniques for Aircraft Control
360 p. 2007 [978-3-540-73718-6]